Ludwig-Maximilians-Universität

Master Thesis

Measuring particle kinematics using template matching algorithms

Author:
Max Weber

Supervisors:
Prof. Dr. Gregor Morfill
Dr. Christoph Räth

Munich, September 29, 2015

Ludwig-Maximilians-Universität

Masterarbeit

Messen der Teilchenkinematik unter Verwendung von Schablonenabgleichs-Algorithmen

Autor:
Max Weber

Betreuer:
Prof. Dr. Gregor Morfill
Dr. Christoph Räth

München, den 29. September 2015

Contents

Chapter 1

Introduction

To gain information about the dynamics of a many particle system, which can in general be any system containing three or more particles, is of great interest in a wide range of fields and especially physics.
One way to gain information about the dynamics is by studying the kinematics of the system, which requires the spatio-temporal knowledge - position and velocity - of every particle in the system for all times. The knowledge of the positions and velocities as a whole enables one to draw conclusions about important characteristic physical properties of the system such as particle density, energy or viscosity.
To draw conclusions about the physical properties of the system one needs an accurate method to estimate the particle positions and velocities and to track them during the time evolution of the system in order to gain information about the particle paths.
An example of such a many particle system is a complex plasma, where a set of colloidal particles is introduced into a charge-neutral plasma [1]. The compelling property of this system is that it can be used as a model system for different states of matter. Depending on the external parameters the complex plasma behaves like for example a solid or a liquid. See [1] for a description of the transition of a plasma crystal from a solid into a liquid phase. But independent of the current state the complex plasma is in, the time evolution of the particles in the complex plasma can be recorded by an optical device, such as a CCD camera, to estimate their kinematics [2], [3].
In order to gain information about the kinematics one has to track all the particles for all time steps to study physical processes, such as for example melting transitions [1], [4]. The information of the time evolution of the particles is usually recorded by an optical system and saved in a sequence of gray value images. While the measurement of the particle tracks itself is not part of this work, we want to focus on the estimation of the particle positions, on which all particle tracking algorithms heavily rely on.
To track a particle efficiently and precisely the tracking algorithm under consideration should use as much information as possible to improve performance. Besides the position, additional information about a particle at a certain time step, such as direction of motion or speed, could be used to improve the results of a tracking

algorithm. Unfortunately this information is usually not known a priori but is a result of the particle tracking algorithm itself.
In this context we present a new method to estimate the speed and direction of motion of a particle out of a single image, whose results can either be used to gain information about the system directly or to support other algorithms with additional information to improve their performance.
If the time evolution of a system is recorded by an optical device, such as a CCD camera, one has to estimate the position and velocity of every particle out of a sequence of images. The conditions under which optical images are recorded are often not ideal and can complicate the position estimation of a particle significantly. Low contrast between objects in the image and the background, scattering in opaque materials or low signal to noise ratio due to high temporal resolution or short exposure time can complicate the task of detecting particles in these images considerably .
For example bright field images may lack contrast, if the features in the image have a low light absorption and therefore appear invisible in the image [5]. Target detection in infrared images can be difficult due to background clutter and low contrast [6],[7]. Living cell imaging, where illumination intensities are reduced to prevent photobleaching and photodamage, is another example of an imaging process with poor contrast [8], [9].
The detection of an object in a gray value image is usually done by associating a high pixel intensity with a center of an object. However, if the contrast in the image is low this can lead to false detections. Therefore the contrast in the image has to be enhanced either by enhancing the structures of interest or by suppressing the noise present.
Several methods have been proposed to fulfil the task of estimating particle positions out of an optical image. Some of the prominent particle detection methods involve techniques like local background subtraction [10], smoothing kernels [11], [12], the combination of local background subtraction and smoothing kernels [13], [14], morphological image filtering [6],[15],[7] and wavelet based detectors [16], [17]. An overview of different spot detection methods can be found in [18].
A quite different approach is the technique of template matching for detecting structures in optical images. Instead of applying operations on the pixel intensities itself, it detects structures by exploiting the statistical similarity of a subregion in an image and an equally sized template containing the structure one is searching for. The technique of template matching is a task of image registration, where it is usually used to detect similar structures in two images and align these images by translation and rotation. Template matching techniques thereby merge the task of feature detection and feature matching, which means finding salient and distinctive objects in an image and finding the correspondence between features in different images. An overview over different image registration techniques, including template matching, is given in [19] and [20].
A crucial part of the template matching algorithm is the similarity measure, which quantifies the amount of similarity between two images. In the context of particle

detection one of the two images is a template, i.e. the structure one is searching for, and the second a sub-region in the recorded image of the system equally sized as the template. The particle positions then correspond to the centers of the subregions in the image with the highest similarity to the template.
As [21] have shown a similarity measure, in their case the Pearsons correlation coefficient (CC), can be used to detect particles in the presence of high noise. However, one drawback of the correlation coefficient is that it only measures linear dependencies between two variables. In contrast the Mutual Information (MI), as a nonlinear measure, does not suffer from this limitation. Therefore we introduce MI as a new similarity measure for the particle detection algorithm via template matching.
Both, the MI and the CC are well known and investigated similarity measures in a wide range of fields. While the CC dates back to 1895, see [22] , [23] for reviews and interpretation, the foundations of MI were developed by Shannon in 1949 [24]. The use of MI for the purpose of image registration was developed both by Collignon [25] and Viola [26]. MI is applied for example in [27] for detecting relationships between variables and in [28] for the task of image registration. For a survey of the use of MI especially for medical images please refer to [29].
One of the main reasons for the use of template matching for particle detection in optical images is its robustness against noise in the image. While [21] show that the CC is very effective in detecting particles in the presence of high Gaussian white noise, the assumption of such a noise distribution is not always valid. In fact there are several examples where the noise is not distributed according to a zero mean Gaussian distribution with a constant standard deviation, but is a function of the signal itself.
Some examples where the noise in optical images is signal dependent are speckle noise in Synthetic Aperture Radar [30] and Ultrasonic Scanner [31] imaging systems. The noise in x-ray CT images is distributed according to a signal dependent Gaussian distribution [32]. The noise in magnetic resonance images follows a riccian distribution [33]. In the case of a CMOS sensor [34] show that intensity dependent photon noise contributes at most to the overall noise variance. A signal dependent noise model of a CCD sensor is presented in [35]. We therefore replace the noise model of [21] by a general signal dependent noise model according to [36].
To access the kinematics of a many particle system, besides the positions also the velocities of the particles have to be estimated. The common way to achieve this is by tracking the particles in consecutive frames and estimating the velocity simply by the quotient of the distance traveled and the time elapsed [37],[38]. This procedure involves the problem of identifying the same particle in consecutive images [10]. This identification problem, commonly referred to as single particle tracking, is present in many scientific fields and there exist therefore many methods to achieve this task.
To name a few examples there is the nearest neighbour approach where two positions of two consecutive frames are associated with the same particle if their distance is minimal with respect to the distances of the particle in the first frame to all the other particles positions in the second frame [13],[14]. Another approach is the combination

of the nearest neighbour method with an additional Kalman filter where one uses a priori knowledge about the system to improve the results of the nearest neighbour method [15]. The seemingly most preferred method is multiple hypothesis tracking [39],[17], which is a global optimization approach taking into account all different particle paths. A comparison of performance about the mentioned methods and other popular particle tracking algorithms can be found in [40].
In the end, all the algorithms have to deal with the problem of the particle identification in consecutive images, which itself relies heavily on a precise particle detection algorithm. If one could estimate the velocity of a particle out of a single image one could overcome the problem related to the identification process or at least supply the tracking algorithm with additional information to improve its results.
At this point our second extension of the template matching technique of [21], which lies in the template itself, comes into play. Where [21] show that the use of a Gaussian kernel as a template can lead to good results in detecting isotropic particles, this kernel is in general not longer applicable for anisotropic shaped particles. The imaging shape of a particle can strongly depend on its velocity. The shape of particles which appear isotropic in a high speed camera could be imaged as an elongated track in a camera with a lower frame rate and increased exposure time. This is due to movement across several pixels during exposure time. The imaged shape of a particle changes from an isotropic to an anisotropic shape and appears elongated in the direction of movement.
While this elongation could lead to serious complications in other particle detection algorithms, the template matching algorithm is not affected by the change of particle shapes. In fact we can take advantage of the information about the elongation of a particle to estimate its position and velocity simultaneously.
To achieve this simultaneous estimation we use a new template design. The new template is the result of mimicking the imaging process of a particle travelling across several pixels during exposure time. With this special template design we are able to estimate how many pixels a particle travels, during exposure time, out of its elongated shape. If one knows the exposure time and the resolution of the image one can link the abstract velocity, which has the unit of pixels per frame, to its physical counterpart.
The problem of tracking particles in different frames to estimate their velocity can then be replaced by estimating the velocity of a particle out of a single image and therefore greatly reduce the complexity of the problem. Even the information about the velocities for all particles in a sequence of images, without any particle tracking, can reveal important physical properties of the system. For example in a complex plasma estimating the velocities of all particles for all time steps out of a series of images enables one to estimate important parameters such as the viscosity of the system [41][42].
The outline of this thesis is as follows. In the second chapter we introduce the methods we use to estimate particle positions and velocities out of an optical gray value image, including the template structures, the similarity measures and the

position and velocity estimation algorithms.
In the third chapter we test our algorithms by applying them to synthetic images. There we start with the synthetic image generation process, followed by position and velocity estimation under different circumstances.
After that we apply our algorithms in the fourth chapter to experimental data coming from the PK-4 experiment, which is currently on the Columbus module on the International Space Station (ISS). In this section we start by introducing the PK-4 experiment and then apply the position estimation algorithm and the velocity estimation method to the experimental data.
As a last point we summarize and comment the results we achieved in this thesis.

Chapter 2

Methods

In this section we introduce the methods we use to detect particles and their velocities in optical images via template matching.
We start by introducing the templates and how to construct them in the isotropic and anisotropic case. After that we go into detail about the two similarity measures we use, their general properties and how to calculate them for two gray value images. We show how to use the similarity measures to detect the particles in a gray value image. As a last part in this section we introduce the method how to estimate a particle's position, velocity and direction of motion simultaneously by template matching.

2.1 Template generation

In order to find objects in a gray value image by template matching one has to construct a template profile which is ideally equal to the intensity profile of the imaged object. As one can imagine, the intensity profiles for different particle shapes differ significantly. Therefore it is crucial for the particle detection by template matching to correctly model the intensity profile of the template, such that it best matches the objects in an image one wants to detect.

2.1.1 Isotropic case

In the case of an isotropic particle we model the intensity profile $I_{x,y}$ of the template to detect it according to [21] as a Gaussian kernel. There are four parameters defining a two dimensional Gaussian kernel, namely the mean and the standard deviation in x and y direction. Due to the finite size of the imaged shape of a particle we calculate the values of the Gaussian kernel inside a window of size $2 \cdot w_x + 1$ in x and $2 \cdot w_y + 1$ in y direction, where the parameters w_x and w_y are integer valued. We always take the Gaussian kernel to be centered at the center of the window such that the free parameters reduce to the standard deviations in x and y direction. The larger the particles we want to detect are, the bigger standard deviations we have to

choose, in order to match the imaged shape of the particles best. The number of rows and columns of the window is taken to be odd, in order to be able to define a center of the rectangle within pixel accuracy.
The intensity values $I_{x,y}$ for the template with standard deviations of σ_x and σ_y can then be calculated by

$$I_{x,y} = I_{max} \cdot \exp\Big[- \big((\frac{x}{\sigma_x})^2 + (\frac{y}{\sigma_y})^2 \big) \Big] \\ x \in [-w_x, w_x], y \in [-w_y, w_y] \tag{2.1}$$

where the values x and y are centered with respect to the center of the window. Here σ_x and σ_y denote the standard deviations in x and y direction and determine the size of the template kernel. Figure 2.1 shows two examples for isotropic templates with a window size 21, i.e. $w_x = w_y = 10$ pixels and a standard deviations of 3 and 5 pixels in x and y direction.

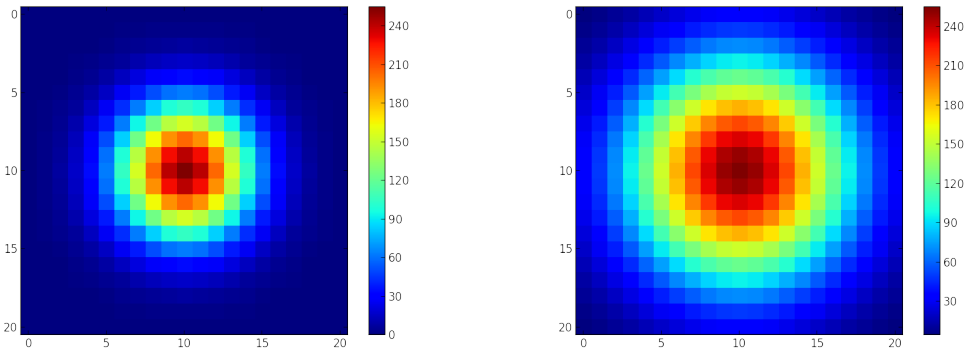

Figure 2.1: Isotropic templates with window size of 21 pixels and standard deviation of 3 (left) and 5 (right) pixels

2.1.2 Anisotropic case

While several imaged shapes of particles can be anisotropic in general, even isotropic shaped particles can change to an anisotropic shape if they move with a high velocity. High velocity here means that it travels a significant amount of distance during exposure time. The particle therefore appears elongated in the direction of motion. To generate a template mimicking a particle with high velocity, one has to take a closer look at the underlying imaging process. Instead of getting imaged at a constant position over expsoure time, the movement creates a shape similar to several particles imaged with constant spatial displacement after each other in the direction of motion. Therefore we simulate the particle movement by adding up several isotropic templates, shifted by equidistant steps Δx and Δy in x and y direction with respect to the preceding template, until we reach the required distance. So for a particle traveling

$N \cdot \Delta x$ equidistant steps in x and $N \cdot \Delta y$ equidistant steps in y direction we construct the corresponding template by following formula

$$I_{elong} = \sum_{i=0}^{N} I_{x+i\cdot\Delta x, y+i\cdot\Delta y} \tag{2.2}$$

where x and y are the center coordinates of the template at beginning of exposure time and $x + N \cdot \Delta x$ and $y + N \cdot \Delta y$ the center coordinates of the isotropic template at the end. In figure 2.2 we show the template which mimics a particle traveling 30 pixels during exposure time. In this case $\Delta x = 1$ and $\Delta y = 0$.

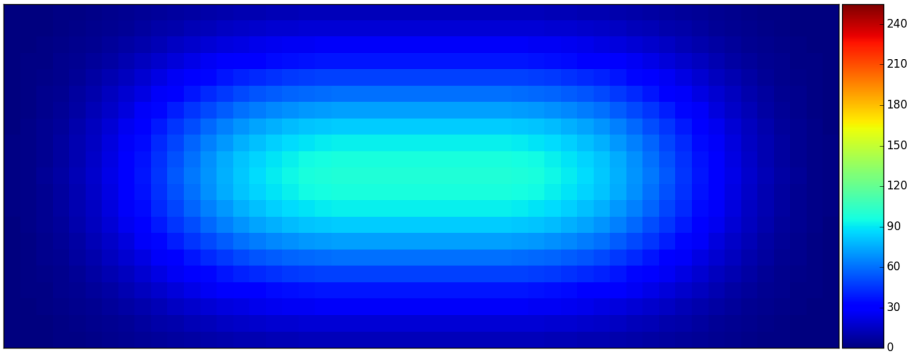

Figure 2.2: Template modeling a particle with velocity of 30 pixels per frame moving in x-Direction

As one can clearly notice the template for the fast particle appears darker than the isotropic template in figure 2.1. While in both cases the scattered amount of light has to be equal during exposure time, the scattered light of the elongated particle is detected over than the scattered light of a isotropic, steady, particle. Therefore the elongated particle is much darker per unit area.
In the case that a particles travels not along a certain axis of the image the situation gets just slightly more complicated. We introduce another parameter ϕ which is the angle between the axis of the elongated particle along the direction of motion and the horizontal axis of the image ranging from $-\frac{\pi}{2}$ to $\frac{\pi}{2}$. The scalar value v becomes a vector quantity $\vec{v}$

$$\vec{v} = \begin{pmatrix} v\cos(\phi) \\ v\sin(\phi) \end{pmatrix} \tag{2.3}$$

In the case of figure 2.2 where ϕ was zero because the movement was along the horizontal axis of the image, the step size in x-direction could be simply set to one and the step size in y direction to zero.
In the case of an arbitrary direction of motion we have to determine a relationship between the step sizes in x and y direction, which we model as a linear relationship.

We make this assumption because the imaging process should not change in the case of an arbitrary direction of motion. From the well known linear equation we can determine the step size in y direction Δy from the step size in x direction Δx given an angle ϕ

$$\Delta y = \tan(\phi) \cdot \Delta x \tag{2.4}$$

In figure 2.3 we show examples of templates all with velocity of 30 pixels per frame in the horizontal direction, but with different values of ϕ and therefore different values of the velocity in the vertical direction.

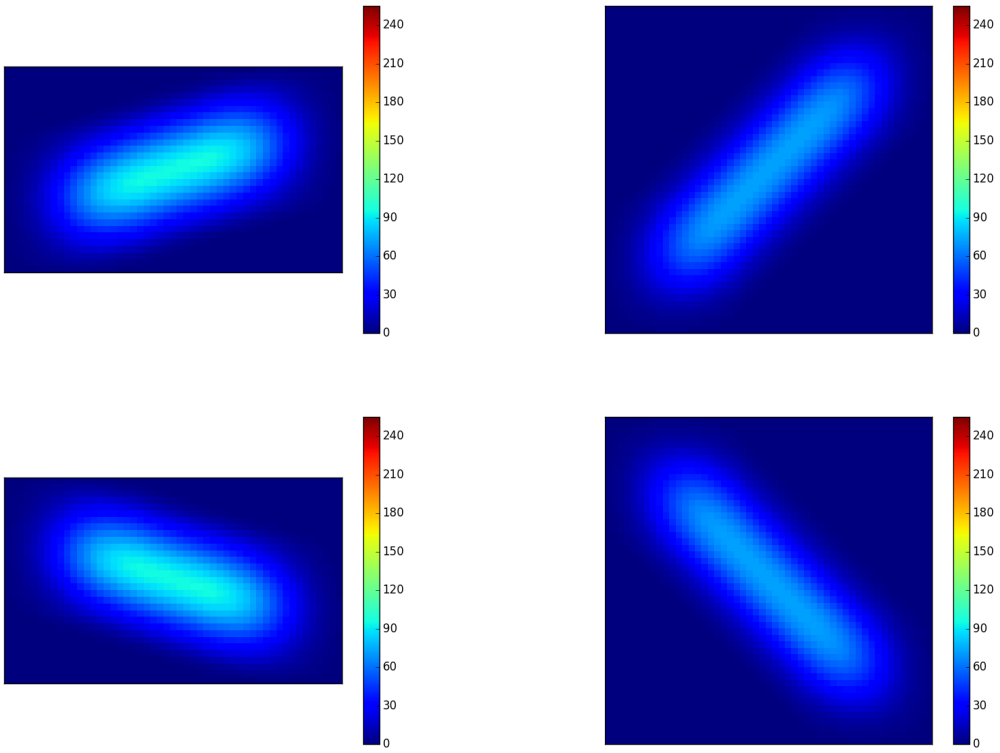

Figure 2.3: Templates for particles with equal velocity of 30 pixels per frame in the horizontal direction, but different directions of motion, left: $\phi = \mp\frac{\pi}{10}$, right: $\phi = \mp\frac{\pi}{4}$ (top row negative, bottom row positive)

2.2 Similarity measures

We consider two different similarity measures for the template matching algorithm, the MI and the CC. In this section we give a formal definition for both similarity measures, list some of the basic properties for both and introduce the methods to calculate them in the context of the template matching algorithm.

2.2.1 Mutual information

The mutual information is in general defined for two random variables, which we will call A and B. For these random variables A and B, their marginal probability distributions $p_A(a)$ and $p_B(b)$ and their joint probability distribution $p_{AB}(a,b)$ we can define the MI by

$$MI(A,B) = \sum_{a,b} p_{AB}(a,b) \log(\frac{p_{AB}(a,b)}{p_A(a)p_B(b)}) \tag{2.5}$$

where we made use of the Kullback-Leibler measure [43].
Two random variables are statistically independent, if the joint probability distribution $p_{AB}(a,b)$ can be rewritten as $p_A(a) \cdot p_B(b)$. In the case of independence the mutual information is zero due to the logarithm of one. Hence the mutual information is measuring the distance between the joint distribution $p_{AB}(a,b)$ and the distribution in the case of independence $p_A(a) \cdot p_B(b)$.
MI is related to the entropies of A and B, $H(A)$ and $H(B)$, and their joint entropy $H(A,B)$

$$MI(A,B) = \ H(A) - H(A|B) = H(A) + H(B) - H(A,B) \tag{2.6}$$

$$H(A) = -\sum_a p_A(a) \log(p_A(a)),\ H(B) = -\sum_b p_B(b) \log(p_B(b)) \tag{2.7}$$

$$H(A,B) = -\sum_{a,b} p_{AB}(a,b) \log(p_{AB}(a,b)) \tag{2.8}$$

Some of the basic properties of the MI are [44]

$$MI(A,B) \geq 0 \tag{2.9}$$

$$MI(A,B) = MI(B,A) \tag{2.10}$$

$$MI(A,A) = H(A) \tag{2.11}$$

$$MI(A,B) \leq \min(H(A),H(B)) \leq \frac{H(A)+H(B)}{2} \leq H(A,B) \tag{2.12}$$

Equation 2.6 gives a good insight about how the mutual information works as a similarity measure. The term $H(A,B)$ can be interpreted as the amount of uncertainty about A and B simultaneously. The more dissimilar A and B are, the bigger $H(A,B)$ becomes and therefore the smaller the mutual information gets. We will demonstrate this by calculating the joint histograms of two random number sequences. We generate the first as a zero mean Gaussian random number sequence with unit standard deviation and add zero mean Gaussian distributed noise with varying standard deviation to this series to generate the second. The amount of dissimilarity between the two sequences is determined by the strength of the applied noise, i. e. its standard deviation. In figure 2.4 we plot the joint histograms for different noise strengths. The values for $H(A)$ and $H(B)$, $H(A,B)$ and the resulting value for the MI are shown below the respective images.

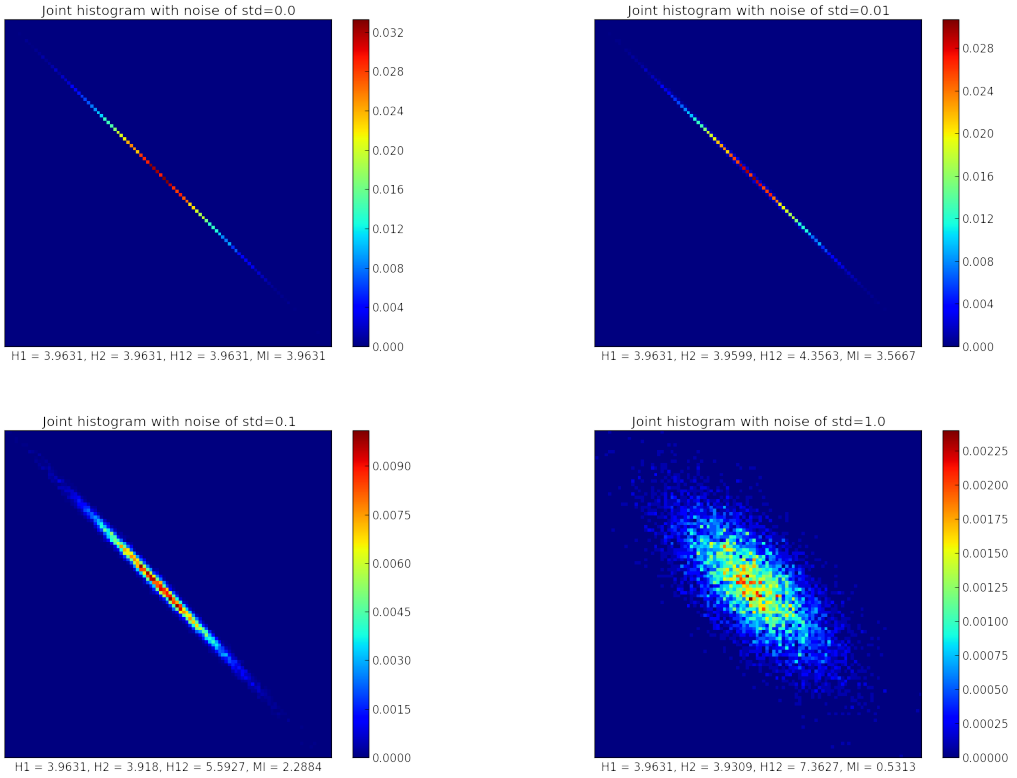

Figure 2.4: Joint histograms for zero mean Gaussian random number series and copies of the series with different noise strengths added. Beneath the images we show the respecting values of the entropies and the MI: Top left: Joint histogram of identical series, Top right: noise added with 1 % strength of the original series standard deviation, Lower left: noise strength with 10% of the original standard deviation, Lower right: noise of equal strength as the original standard deviation

If we think of the elements of two equally sized gray value images A and B as random variables we can calculate the MI for these values. We approximate the marginal probability distributions $p_A(a)$ and $p_B(b)$ and the joint probability density $p_{AB}(a,b)$ by binning gray values of the two images and calculate their histograms normalized by the number of elements N in the images.

$$p_A(a) \approx \frac{1}{N}\sum_{i=1}^{N}\theta(A_i,a),\; p_B(b) \approx \frac{1}{N}\sum_{j=1}^{N}\theta(B_j,b) \tag{2.13}$$

$$p_{AB}(a,b) \approx \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N}\theta(A_i,B_j,a,b) \tag{2.14}$$

Where $\theta(A_i,a)$ is the rectangle function which evaluates to one if the element A_i is in bin a and to zero if not. Similar for $\theta(B_j,b)$ and $\theta(A_i,B_j,a,b)$.
See [45] for a review of the different estimation methods for the MI. If using MI as

a similarity measure, one has to be aware of the fact that the result is dependent on the size of the image. By that we mean that if one calculates the MI of one image with itself the resulting value is higher for bigger images. Therefore we use a normalized form of the MI, denoted by $\widehat{MI}(A,B)$,which takes a maximum value of one independent of the image size [46]

$$\widehat{MI}(A,B) = \frac{I(A,B)}{\sqrt{H(A) * H(B)}} \tag{2.15}$$

As we will always use the normalized form of the MI in the following, we will use the same abbreviation, MI, for it as before for the unnormalized form.

2.2.2 Pearson Correlation Coefficient

In general for two random variables A and B the CC is defined by

$$\rho(A,B) = \frac{\mathrm{Cov}(A,B)}{\sigma(A)\sigma(B)} \tag{2.16}$$

where Cov denotes the covariance of the random variables A and B and $\sigma(A)$ and $\sigma(B)$ their standard deviation, respectively. Some of the main properties of the CC are [23]

$$\rho(A,B) \in [-1,1] \tag{2.17}$$
$$\rho(A,A) = 1 \tag{2.18}$$
$$\rho(A,-A) = -1 \tag{2.19}$$
$$\rho(A,B) = 0 \Leftrightarrow A \perp\!\!\!\perp B \tag{2.20}$$

Again thinking of the elements of A and B as a a sequence of realizations of two random variables, the correlation coefficient ρ of two gray value images of equal size N is defined by

$$\rho(A,B) = \frac{1}{N_i N_j} \sum_{i=1}^{Ni} \sum_{j=1}^{Nj} \frac{(A_{ij} - \mu_A)(B_{ij} - \mu_B)}{\sigma_A \sigma_B} \tag{2.21}$$

where N_i is the number of rows and N_j the number of columns of the images, μ_A and μ_B are the mean values of the image gray values and σ_A and σ_B their standard deviation respectively. A_{ij} and B_{ij} denote the values at position (i,j).

2.3 Particle detection algorithm

In this section we review the method for detecting particles in a gray value image by template matching. The method can be divided into two main steps.
The first is to calculate the similarity measure under consideration for the whole

gray value image. We obtain the similarity measure for every pixel in the image by calculating the similarity value of a sliding window around every pixel in the image with the template and save the resulting value in a new matrix, which has the same number of rows and columns as the image. We will call this new matrix the similarity measure map in the following. The sliding window has to have the same size as the template. So every pixel in the similarity measure map corresponds to the similarity value between a sub-region in the image around the pixel and the template.
Detecting the particles is then done by finding the local maxima in the similarity measure map. Here, we adapt the method from [13], we search for the largest maximum in the similarity measure map, set a region around this local maximum to zero and search again for the next local maximum until the new local maximum falls below a threshold we choose. These local maxima correspond to the coordinates with the highest local similarity to the template and therefore to the most probable particle centers according to the template matching algorithm.
The assumption of associating the local maxima with the particle centres produces good results in the case of low noise. However, in the case of high noise this assumption may lead to errors and an improved method is needed to account for the fact that due to noise the local maximum does not correspond to center of a particle, but misses it by some pixels.
One method to improve the results is by performing a so called weighted mean, sometimes also referred to as center of mass, calculation around the local maxima to compensate for the error induced by the noise. Given the position of a local maximum we first choose the size of a region around that maximum, which is typically of the order of one or two pixels, and then calculate the coordinates of the weighted mean in this region, denoted by $\bar{x}$ and $\bar{y}$, by

$$\bar{x} = \frac{\sum_{i=1}^{N} x_i * v_i}{\sum_{i=1}^{N} v_i}, \bar{y} = \frac{\sum_{i=1}^{N} y_i * v_i}{\sum_{i=1}^{N} v_i} \tag{2.22}$$

where N are the number of values in the region around the local maximum and v_i denotes the ith value of the region in the similarity measure map. This method can also be used to achieve subpixel accuracy.
In figure 2.5 we illustrate the whole process. On the left, we show the original noisy image with the template we use for the particle detection in the upper left corner of the noisy image. In the middle, we show the similarity measure map, calculated via MI, for the noisy image and the template. On the right, we plot the local maxima of the similarity measure map, which correspond to the estimated centers according to the method, on top of the noisy image.

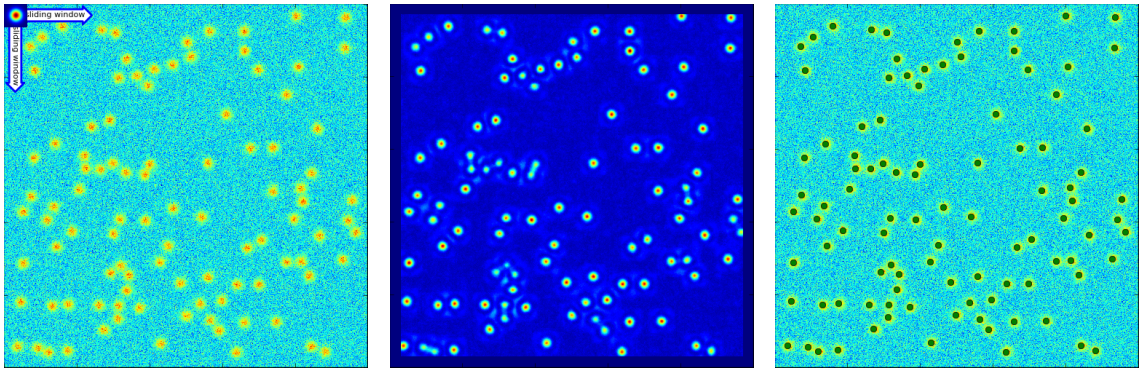

Figure 2.5: Illustration of the particle detection algorithm: left: original noisy image; middle: similarity measure map; right: original noisy image with estimated centres plotted as green dots

For this illustration we used a synthetic image with 100 isotropic particles with standard deviation of 5 pixels, signal dependent noise and a signal to noise ratio of 5. For synthetic images the true particle locations are known. We therefore can compare the estimated positions to them and thereby measure the performance of the similarity measures. We will consider a particle as truly detected with accuracy δ if the estimated position and the true position does not differ more than δ. We define δ by the euclidean distance

$$\delta = \sqrt{(x_{true} - x_{est})^2 + (y_{true} - y_{est})^2} \tag{2.23}$$

Here x_{true} and y_{true} denote the known real x an y coordinate of the particle in the synthetic image and x_{est} and y_{est} the x and y coordinates estimated by the particle detection algorithm.

2.4 Positions, velocity and direction of motion estimation

In the case of estimating the position, speed and direction of motion of a particle out of an image, the dimensionality of the problem increases. In the case of the pure position estimation one has to calculate the similarity measure map once for every pixel in the image. Now in the case of the simultaneous estimation of position, velocity and direction of motion the dimension of the similarity measure map increases. One has to calculate the similarity measure map with a template for each different velocity and different direction of motion for every pixel in the image. So for a image of R rows and C columns the resulting similarity measure map has the dimension of $R \times C \times K \times M$ for K different velocities and M different directions of motion.
To find the maximum values according to position, velocity and direction of motion we first determine the maximum similarity measure map value with respect to the parameters velocity and direction for every position in the image, independent of

whether it is the estimated center of a particle or not. We thereby reduce the dimension of the similarity measure map to $R \times C$ and call it the reduced similarity measure map in the following. Afterwords we calculate the positions of the particles out the reduced similarity measure map the same way as before for the pure position estimation.
We demonstrate the whole process we described in this section in figure 2.6. We generate a noisy synthetic image containing a single particle with velocity of 30 pixels per frame and an angle of motion of $0.1 \cdot \pi$ with respect to the horizontal axis of the image. In figure 2.6 we show the synthetic image in the center of the figure. On the top row we show the reduced similarity measure map calculated by MI on the left and by the CC on the right. On the lower row we show the values for the similarity measure maps for different angles and velocities at the particle position calculated from the reduced similarity measure maps. The estimated centers are plotted in the reduced similarity measure maps and in the image as green dots.

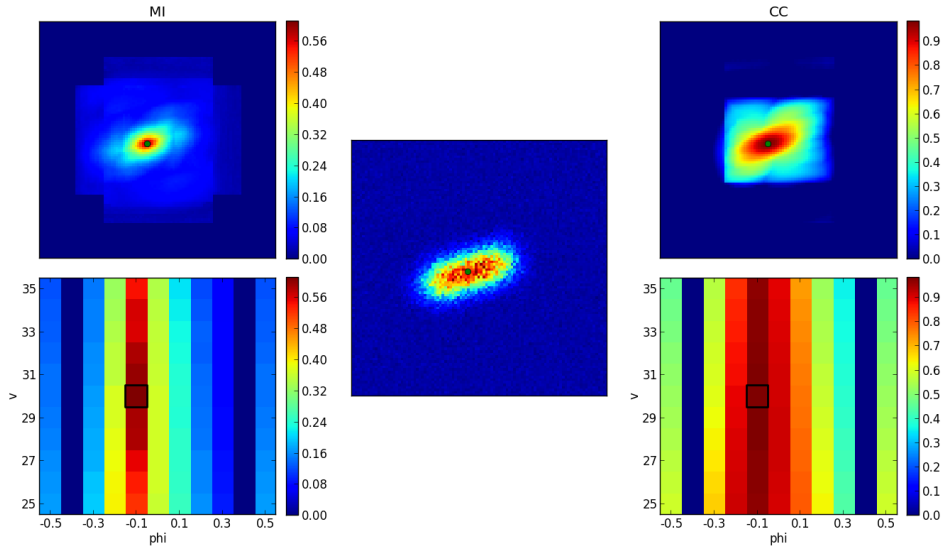

Figure 2.6: Illustration of the position, velocity and direction of motion estimation algorithm: Top row: reduced similarity measure maps; center: original noisy image containing one noisy particle; bottom row: similarity measure values for different velocities and angles of motion at the estimated position of the particle

The maximum value at the estimated center position, which corresponds to the estimated velocity and direction of motion, is outlined by a black frame.

Chapter 3

Synthetic images

To accurately test the particle detection algorithm performance we use synthetic images, where we know the positions, velocities and directions of motion for every particle in the image. In this section we give a short overview over the synthetic image generation process including the signal independent and signal dependent noise models.
We apply our particle detection algorithm to the synthetic images under different noise conditions and different particle densities to measure the performance of both similarity measures. After that we investigate the accuracy of the similarity measure in the estimation of velocity and direction of motion for different noise strengths and types. As a last part, we show a typical application of the velocity estimation procedure where we estimate the averaged velocity profile out of synthetic images.

3.1 Image generation

The generation of the synthetic images involves the following steps. First we generate random positions for the particle centers, taking into account a minimal distance of the particles. Then at each random position we add a rectangular box containing a particle in the center. We generate the imaged shape of a particle the same way as we model the templates. After adding the particles to the image we corrupt the image with additive Gaussian distributed noise denoted by η with standard deviation of one and multiply it by σwhich characterizes the strength of the noise.

$$I = I_{orig} + \sigma \cdot \eta \tag{3.1}$$

After that we normalize it again to the range from 0 to I_{max}, where I_{max} is the maximum value of the uncorrupted image I_{orig}. The strength of the noise σ is defined by the signal to noise ratio (SNR), which is a free parameter in the image generation process.

$$SNR = \frac{I_{max}}{\sigma} \tag{3.2}$$

3.2 Noise model

As we mentioned in the introduction, we are considering not only signal independent, but also signal dependent image noise. To model the noise in the synthetic images we adapt the general linear noise model of [36], which is in general a function of the noise free image I_{orig}:

$$\eta(I_{orig}) = \sqrt{I_{orig}} \cdot \xi_{dep} + \xi_{ind} \tag{3.3}$$

Where ξ_{dep} and ξ_{ind} are both zero mean, mutually independent, Gaussian white noise fields.

$$\begin{aligned} \xi_{ind}^{i,j} &\sim \mathcal{N}(0,1) \\ \xi_{dep}^{i,j} &\sim \mathcal{N}(0,1) \end{aligned} \tag{3.4}$$

The term $\sqrt{I_{orig}} \cdot \xi_{dep}$ is the signal dependent part of the noise model. It models the contribution of the so called shot noise, which is present in many imaging scenarios. The term ξ_{ind} models signal independent background noise in the image. To test the dependence on the different noise components we introduce two additional parameters α and β, both either 0 or 1.

$$\eta = \alpha \cdot \sqrt{I} \cdot \xi_{dep} + \beta \cdot \xi_{ind} \tag{3.5}$$

The noise array is always normalized to a standard deviation of one before it is multiplied by σ and added to the noise-free image. This is necessary because we define the overall noise strength σ by the SNR (see 3.2). If the noise had a standard deviation larger than 1 we would in fact add stronger noise to the image than the amount corresponding to the SNR.

3.3 Particle detection in the presence of strong noise

In this section we compare MI and CC for different noise strengths and noise compositions. The performance of each measure is assessed by the number of true positive detections compared with the number of false detections. In addition we compare the detection accuracy in terms of mean detection error and standard deviation of the detection error as a function of the SNR for the different noise compositions.

3.3.1 Signal independent noise

In this section we investigate the performance of the MI and the CC in the pure signal independent noise regime. We compare the detection rates of true positive and false positive detections for different SNR values and detection accuracies δ. We create the synthetic images for the performance evaluation with the following parameters. 100 isotropic particles with standard deviation of 5 pixels are distributed

randomly in the images. The noise strengths vary from a SNR of 10 to a SNR of 0.5 and we calculate the performance of the similarity measures for accuracies δ of 0, 1 and 2 pixels.
The results for the MI and the CC are shown in the following figures, where we plot the number of true positive detections (tp) in the upper row and the number of false positive detections (fp) in the lower row. Both are plotted against the threshold for the maximum detection algorithm, which we vary between half of the maximum of the similarity measure maps, denoted by h_{MI} and h_{CC} respectively, to the maximum in 20 equidistant steps denoted by Δ_{MI} and Δ_{CC} respectively. We begin with a detection accuracy of $\delta = 0$.

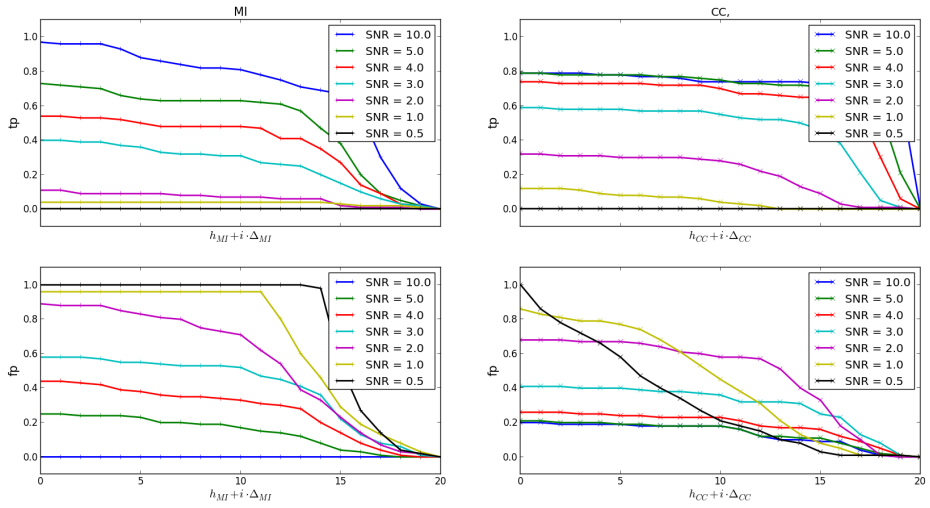

Figure 3.1: tp (upper row) and fp (lower row) rates for signal independent noise, $\delta = 0$

For this accuracy MI outperforms the CC for a SNR of 10. For the SNR values of 5 to 1 the CC outperforms the MI while for a SNR value of 0.5 both similarity measures perform very poorly.
Now we lower the detection accuracy to one pixel, allowing that also detections one pixel around the true particle center to be counted as true positive detections.

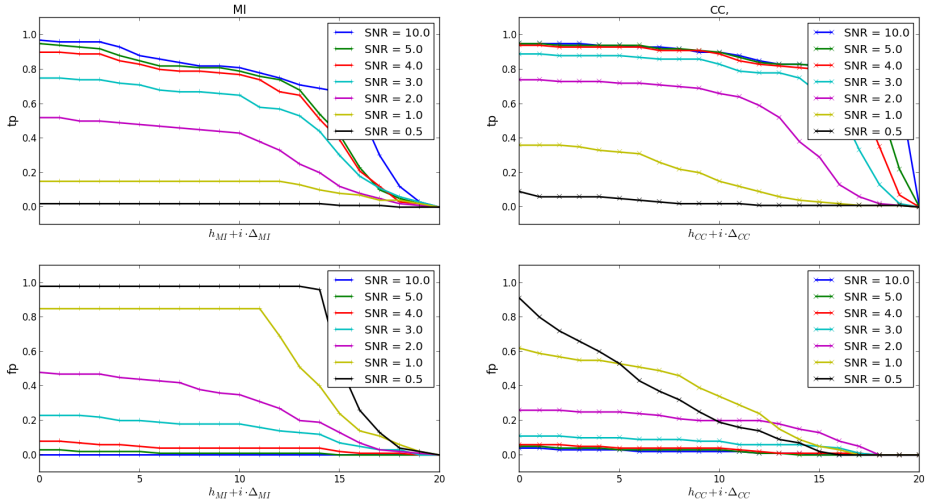

Figure 3.2: tp (upper row) and fp (lower row) rates for signal independent noise, $\delta = 1$

Both similarity measures perform substantially better. Now more than half of the particles are detected correctly by the MI at a SNR of 2. The CC even detects nearly 40% of the particles for a SNR of 1 correctly.
If we lower the detection accuracy further, and count detections with a maximal distance of two pixels as true positive positions detections, we achieve following detection rates shown in figure 3.3.

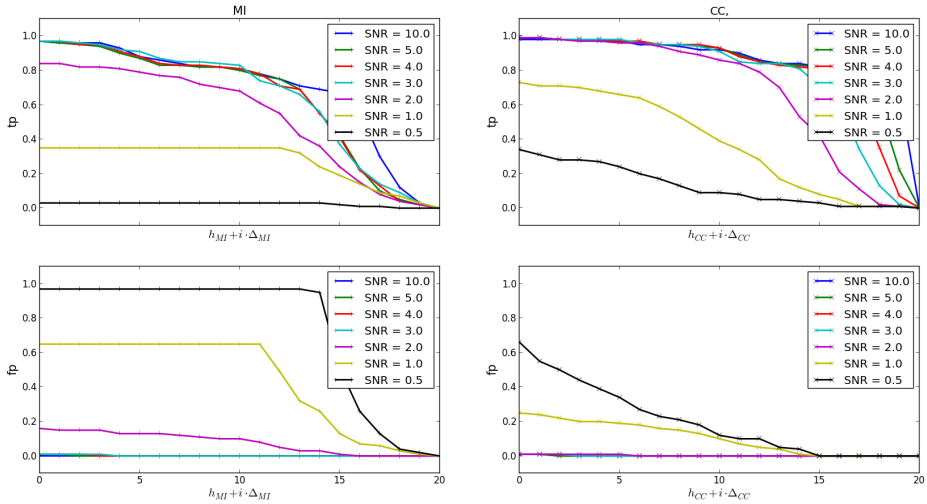

Figure 3.3: tp (upper row) and fp (lower row) rates for signal independent noise, $\delta = 2$

We again observe an improvement in performance for both similarity measures. Still the CC outperforms the MI for SNR values of 2 and below. For the CC we can

claim that we can still detect more than 80 % of the particles correctly up to a SNR of 1, if δ is not required to be higher than two.
Overall one can say that in the signal independent regime the CC performs significantly better for low SNR values than the MI while for $\delta = 0$ MI performed better for the high SNR values.
Now we show the mean detection accuracy δ as a function of the SNR value for signal independent noise. We calculate this by estimating the position of one particle 100 times, calculate the detection error between the known particle position and the estimated particle position by 2.23 and finally calculate the mean value and standard deviation of the detection accuracy for the 100 particles. The parameters for the particles are an isotropic shape with standard deviation of 5 pixels in a box of 31 pixels. For every particle we estimate the position by searching for 1 particle in the similarity measure map. Due to the choice of the threshold for the particle detection algorithm of 0.1 times the maximum value of the similarity measure map, it can happen that detections occur with a very large deviation from the true position. In this case the similarity measure of use "sees" only noise and detects noisy subregion of the image by chance as a particle. Unfortunately there is no way that one could make a threshold independent of the dynamic range of the similarity measure maps, as the dynamic range highly depends on the noise strength. Therefore we disregard detections with a distance of 15 pixels, half of the size of the box, and larger from the true particle positions for detection accuracy method. To give an idea of the amount of detections purely due to noise we plot the total number of detections with a distance larger as 15 pixels, which we denote by ϵ, separately.
In the following figure we plot δ and ϵ both as a function of the SNR value from 0.5 to 10 for pure signal independent noise.

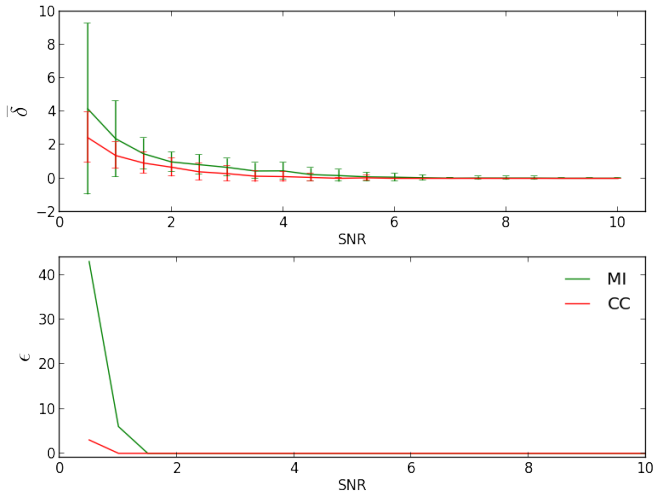

Figure 3.4: mean detection accuracy δ and number of detections due to noise ϵ for signal independent noise

Here we observe a similar performance of the MI as for the true positive rates.

The error in position estimation is significantly smaller for low SNR values in the case of the CC, compared to the MI. The MI also detects many particles falsely due to noise for a SNR value of 0.5.

3.3.2 Signal dependent noise

In the pure signal dependent noise regime we perform the same procedure as for the signal independent. We test the particle detection algorithms with synthetic images of 100 isotropic particles with standard deviations of 5 pixels. The SNR in the images ranges from 10 to 0.5 and we test the algorithms for a the detection accuracies of 2, 1 and 0.
In figure 3.5 figure shows the results for $\delta = 0$.

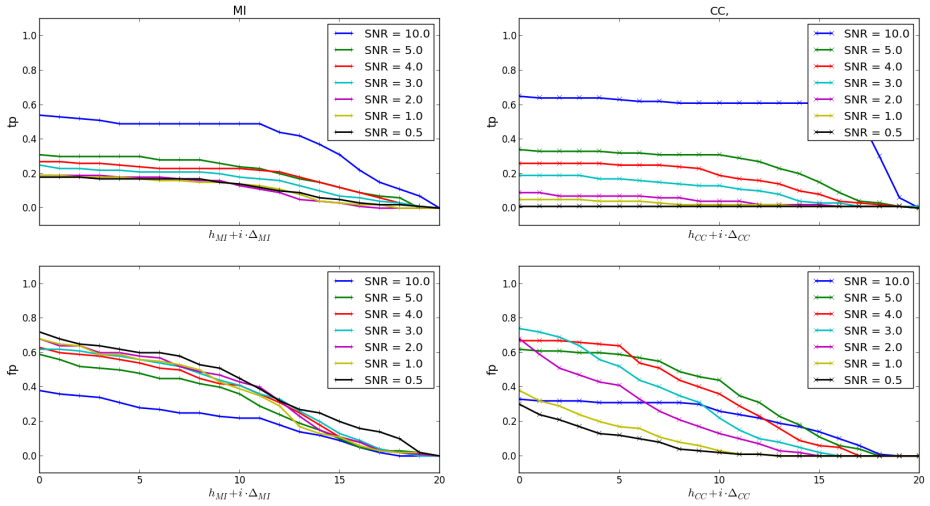

Figure 3.5: tp (upper row) and fp (lower row) rates for signal dependent noise, $\delta = 0$

For a SNR value of 10 the CC outperforms the MI. For a SNR value of 5 the performance of both similarity measures is nearly equal and for SNR values below 5 the performance of the MI is almost stable, while the true positive rate of the CC decreases to zero.
If we lower the detection accuracy δ to one pixel we get the results shown in figure 3.6

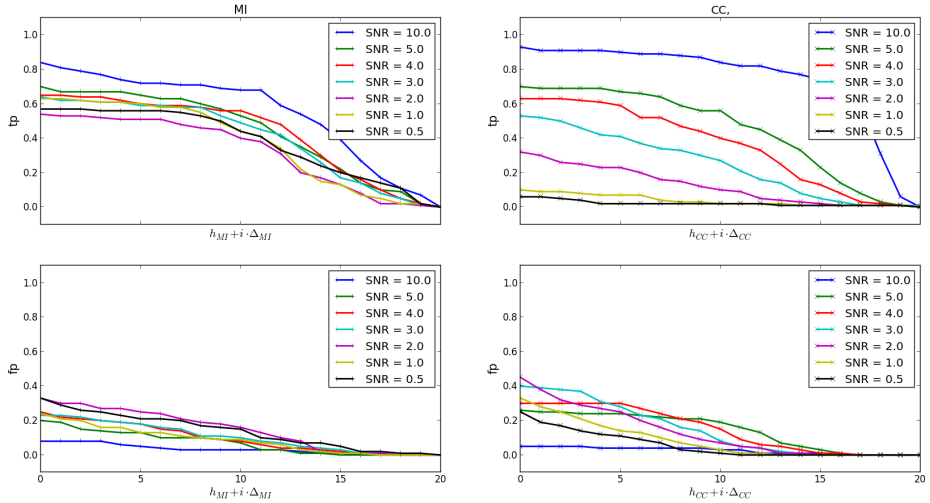

Figure 3.6: tp (upper row) and fp (lower row) rates for signal dependent noise, $\delta = 1$

We can see that both similarity measures perform better overall.
At at a SNR of 3 both perform nearly equally good and for SNR values below 2 the performance of the CC decreases to zero while the true positive rate of the MI is nearly constant for all SNR values including 4 and below
Finally we lower the detection accuracy further to two pixels.

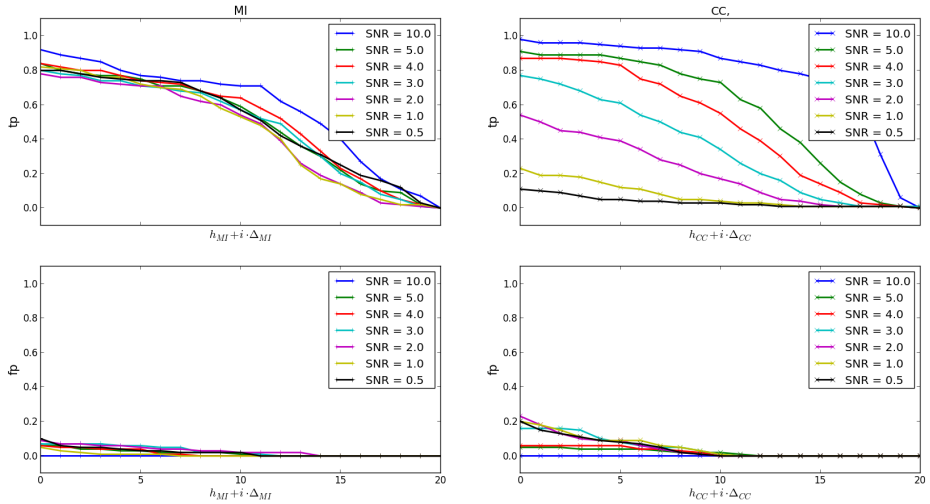

Figure 3.7: tp (upper row) and fp (lower row) rates for signal dependent noise, $\delta = 2$

The performance of both similarity measures improves further. While the performance of the CC is below 0.4 for a SNR of 1 and decreases further below 0.2 for a SNR of 0.5 the detection rate in the case of the MI is nearly unaffected for this

accuracy.
In this noise regime we show that MI is able to detect a high amount of particles, nearly independently of the SNR for $\delta = 2$, while the performance of the CC suffers significantly for low SNR values.
In the figure 3.8 we show the dependence of δ and ϵ to the SNR for pure signal dependent noise.

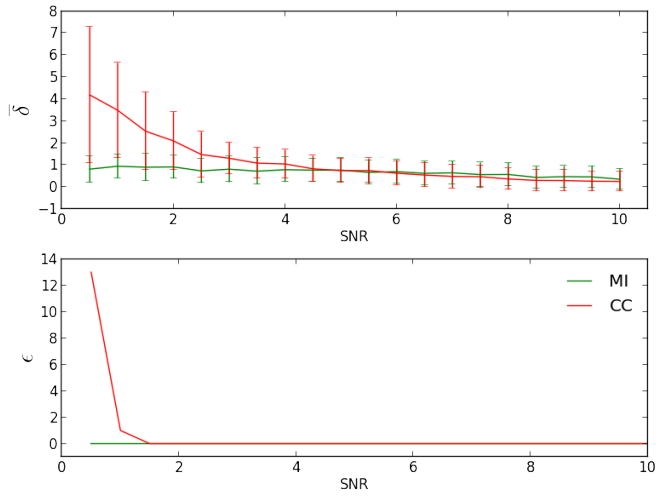

Figure 3.8: Detection accuracy δ and number of detections due to noise ϵ for signal dependent noise

Here we observe quite the opposite behaviour of the performance compared to the signal independent noise regime. While the error in case of the MI is very low and nearly constant for all SNR values the CC has a very high error in the low SNR regime.

3.3.3 Signal independent and dependent noise

In this noise regime we again perform the same evaluation as in the pure signal independent and in the pure signal dependent noise regimes for the MI and CC. We use synthetic images with 100 isotropic particles with standard deviations of 5 pixels and varying SNR values from 10 to 0.5 and detection accuracy values δ of 0, 1 and 2 pixels. The results for a detection accuracy 0 are shown in figure 3.9.

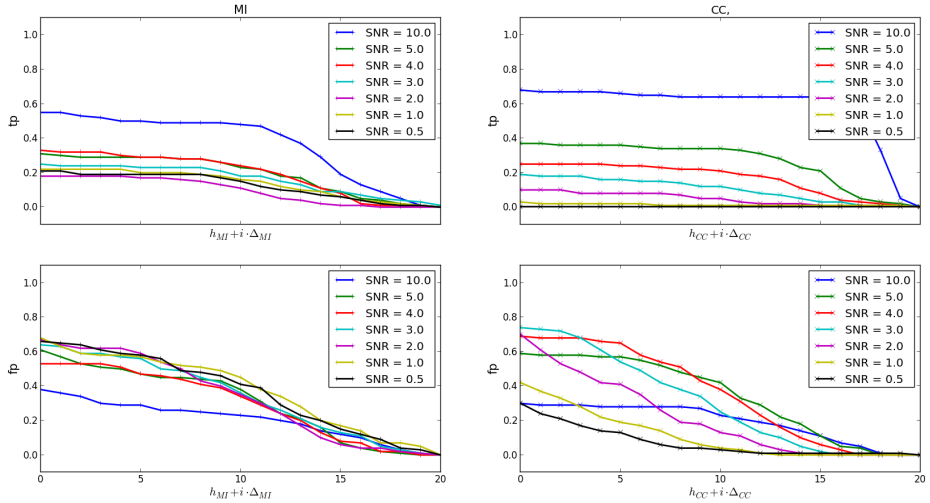

Figure 3.9: tp (upper row) and fp (lower row) rates for signal dependent and independent noise, $\delta = 0$

As in the pure signal dependent noise regime, the CC outperforms the MI for a SNR value of 10. For a SNR value of 5 their performance is nearly equal. Again for SNR values including 4 and below the performance of the MI is almost constant, while the tp rates of the CC decrease to zero for increasing noise.
The performance of both similarity measures for $\delta = 1$ is shown in figure 3.10.

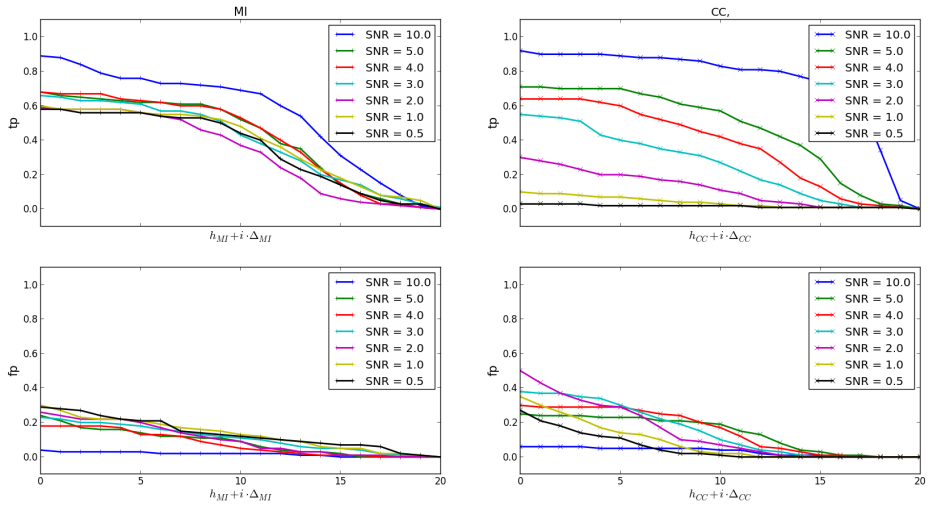

Figure 3.10: tp (upper row) and fp (lower row) rates for signal dependent and independent noise, $\delta = 1$

The results for the MI improve substantially for all SNR values and the perfor-

mance again is almost constant at a certain SNR value of 4. Also the performance of the CC improves but is still below 0.5 for a SNR of 2, while Mi performs better than 0.5 for all SNR values.
For a detection accuracy of two the results are shown in figure 3.11.

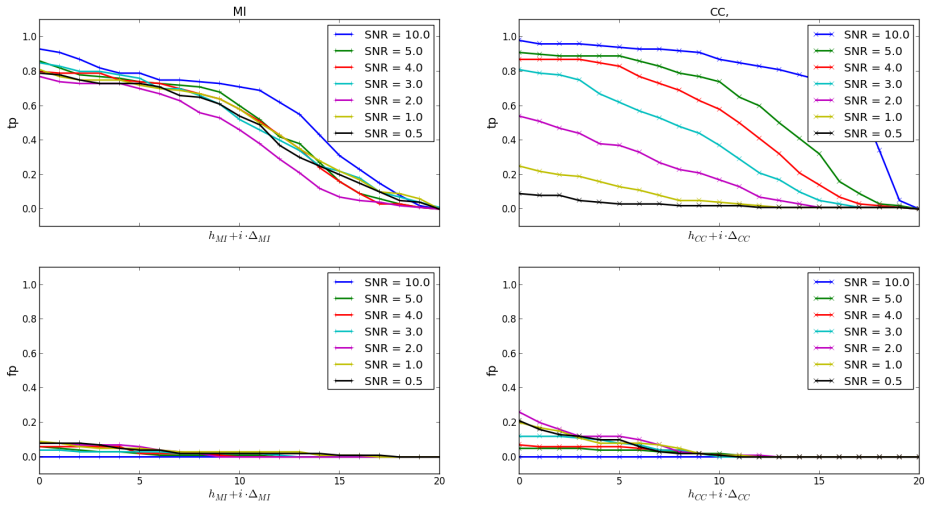

Figure 3.11: tp (upper row) and fp (lower row) rates for signal dependent and independent noise, $\delta = 2$

Similar to the pure signal independent noise regime the performance of the MI is affected very little by decreasing SNR values for this noise regime. In the case of the CC the performance breaks down at a lower SNR value of 1.
In this noise regime we show that MI is able to detect a high amount of particles nearly independently of the SNR if the minimum detection accuracy of two pixels is sufficient. In general we can claim that CC is not suited for particle detection in this regime and for a low SNR value. Again we plot δ and ϵ as a function of the SNR.

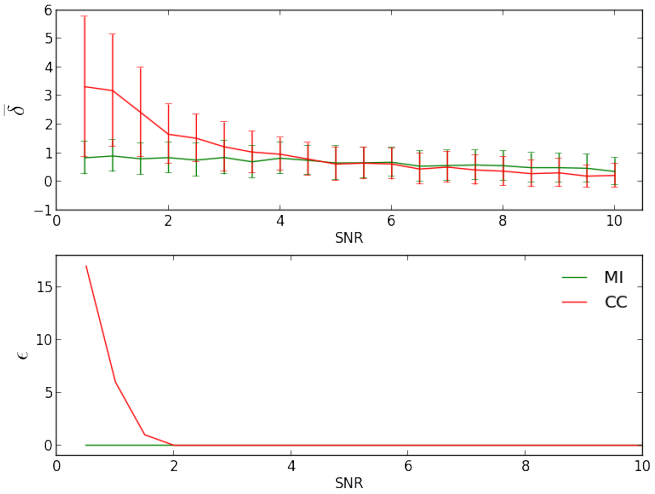

Figure 3.12: Detection accuracy δ and number of detections due to noise ϵ for signal independent and signal dependent noise

For a mixture of signal independent and signal dependent noise the performance in detection accuracy is very similar to the pure signal dependent noise regime. The MI clearly outperforms the CC both in accuracy and detections due to noise.

3.4 Dependence of the similarity measures to the interparticle distance

In case of dense particle clouds, detecting the particles can be difficult due to overlapping imaged shapes of nearby particles. We therefore investigate the dependence of the particle detection performance of the MI and CC in the case of high density particle clouds using synthetic images containing five particles. The particles in the synthetic images are all modelled by the same equally shaped isotropic Gaussian kernel with a standard deviation of five pixels in a quadratic box of 31 pixels. The five particles in the images we use to test the performance of the similarity measures are arranged in the same order in all the images, but with varying peak to peak distances.
We position one particle in the center of the image and the four other particles around the center particle, all with the same distance to it. The minimal distance of the particles in the images is defined by the distance from the middle particle to the outer particles. Although the impact of the noise is not in the focus of this investigation, we add some noise with a SNR of 20 to the images to create a more realistic environment.
In the following figures we show in the first column the noisy image in the upper, the corresponding similarity measure map calculated by MI in the middle and calculated by CC in the lower row. The second column shows a horizontal and the third a

diagonal slice through the image and the similarity measure maps respectively.
The performance of the two similarity measures is most clearly seen in the horizontal and diagonal slices of the respective similarity measure maps which we plot together with the slices through the original image without noise. We show the slices through the MI in green, through the CC in red and through the original image in blue. The slices of the original image are normalized to the maximum of the respective similarity measure maps for demonstration purposes. Ideally, the similarity measure maps should show a single peak in the second column, which corresponds to the center particle, and three distinct peaks for the diagonal slice. We only show one diagonal slice as the whole system is point symmetric.
In figure 3.13 we show the performance for the case of non overlapping particles.

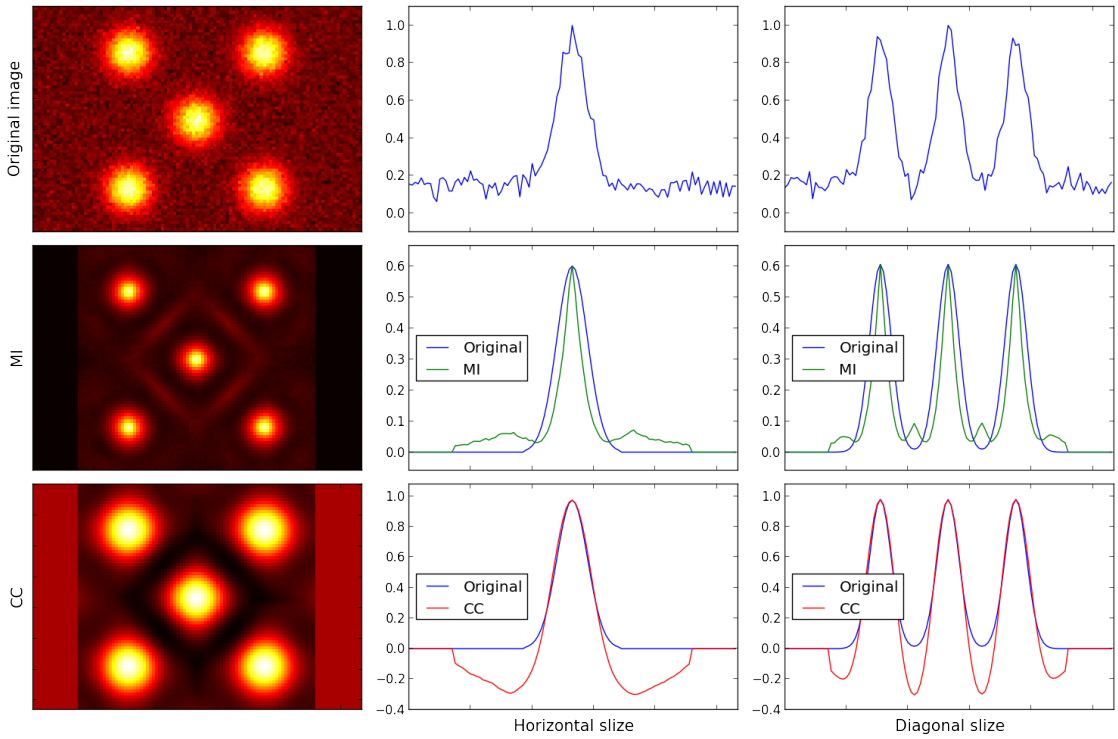

Figure 3.13: Response to synthetic noisy image of 5 particles (upper row) of the MI (middle row) and the CC (lower row) for minimum peak to peak distance of 30 pixels

For this minimal distance both similarity measures detect all the particles correctly, as one would expect given that there are no major complications as high noise in the image.
Now we decrease the minimum distance of the particles to approximately 18 pixels.

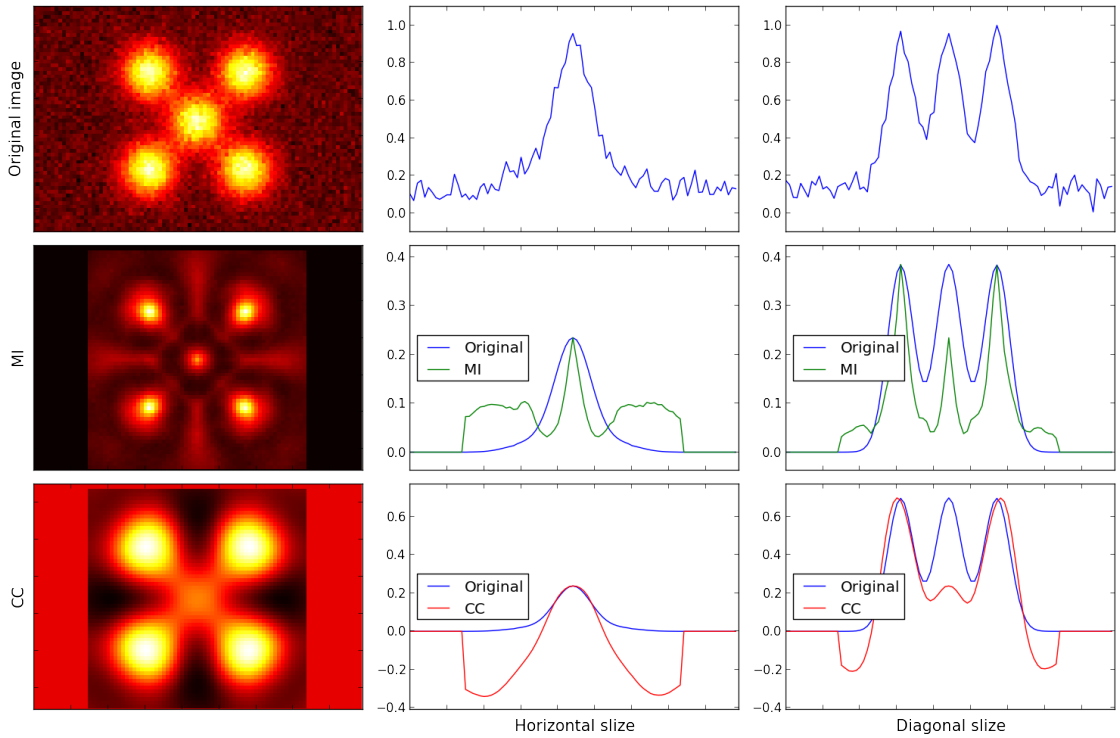

Figure 3.14: Response to synthetic noisy image of 5 particles (upper row) of the MI (middle row) and the CC (lower row) for minimum peak to peak distance of 18 pixels

Still both similarity measure separate the particles correctly. However, one can observe that in the case of the CC the peak corresponding to the center particle starts to vanish and is significantly below the peaks corresponding to the outer particle centers in the diagonal slice. If one would choose the minimal distance in the maxima detection algorithm to low this peak would not be detected, but a fifth particle would falsely be detected at a position near one of the outer particles.
In figure 3.15 we decrease the minimal distance further to approximately 14 pixels.

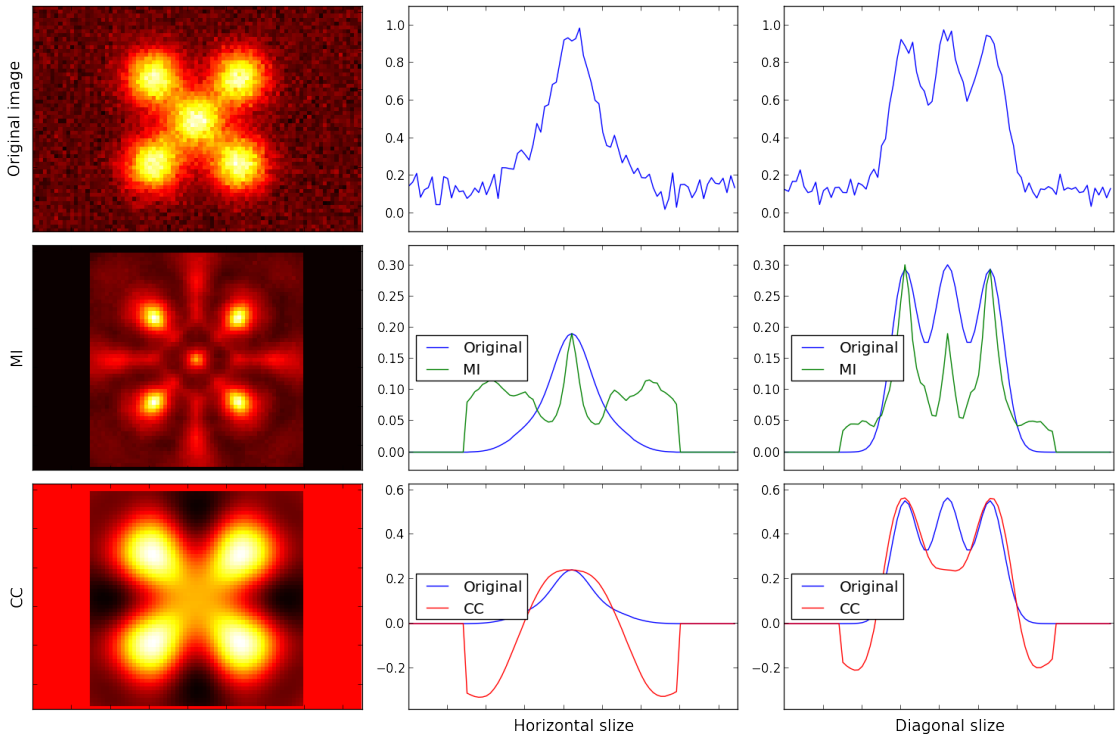

Figure 3.15: Response to synthetic noisy image of 5 particles (upper row) of the MI (middle row) and the CC (lower row) for minimum peak to peak distance of 14 pixels

For the CC the middle peak in the diagonal slice vanished. Now depending on how one chooses the parameters for the maxima detection, either only four particles would be detected, or a fifth particle would be detected falsely near the outer particles. In contrast the MI shows still three distinct peaks in the diagonal slice.
As a last example we show the results for a minimal distance of 12 pixels.

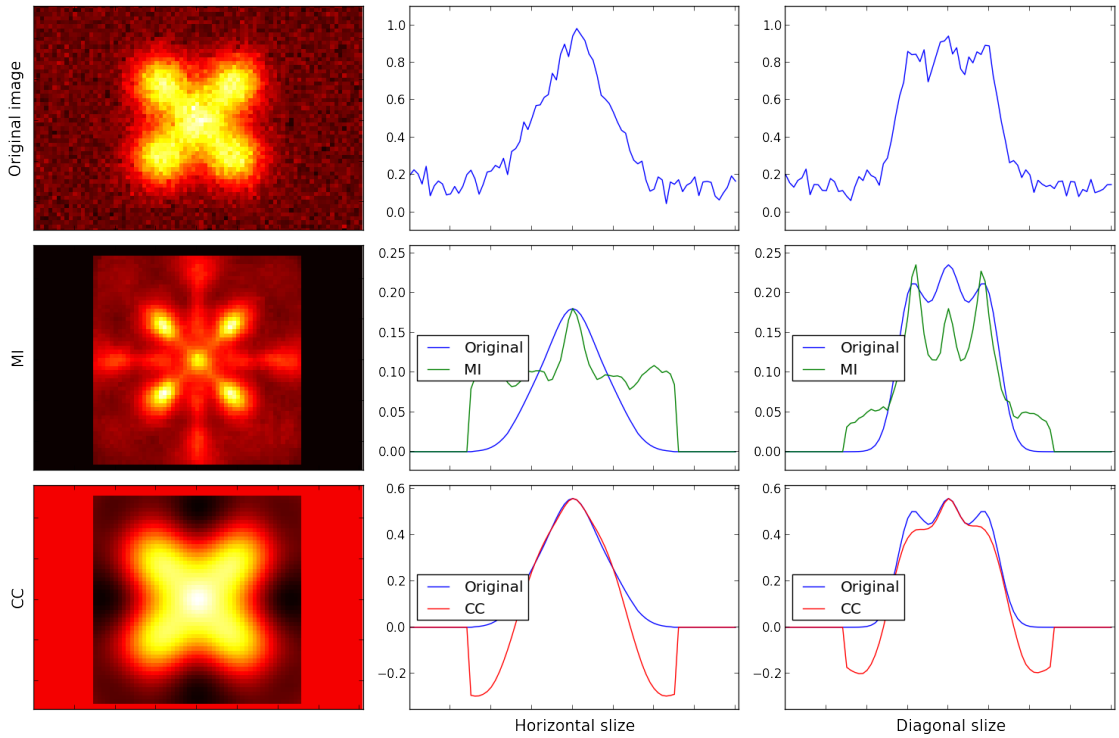

Figure 3.16: Response to synthetic noisy image of 5 particles (upper row) of the MI (middle row) and the correlation for minimum peak to peak distance of 12 pixels

Here the results for the CC get even worse. All of the five particles, which can be still separated by the MI, cannot be separated any more by the CC. The MI in contrast separates the particles at the nearly exact true positions.
The capability of the MI to separate particles even in the case of a low interparticle distance becomes crucial if one tries to locate particles in a dense three dimensional cloud. Observed with one camera, all of the particles are projected on to the imaging plane of the camera. Particles which are originally behind each other along the axis perpendicular to the imaging plane can reduce the interparticle distance substantially and even overlap in the camera image.
However, if the MI is used to detect particles in images with low interparticle distance one has to deal with an intrinsic property of the MI itself. While the CC assigns positive values to correlated images and negative values to anti-correlated images, the MI cannot distinguish between those two situations. It will assign the same value to the images independent of a positive or negative correlation. This can lead to false detections, caused for example by an empty space in the image surrounded by particles. We show this effect by the same setup as before with only one difference. We leave out the middle particle creating an empty space in the middle of the four outer particles. We choose the same values for the minimal peak to peak distance as before.
We begin by showing the case of a minimal peak to peak distance of 30 pixels.

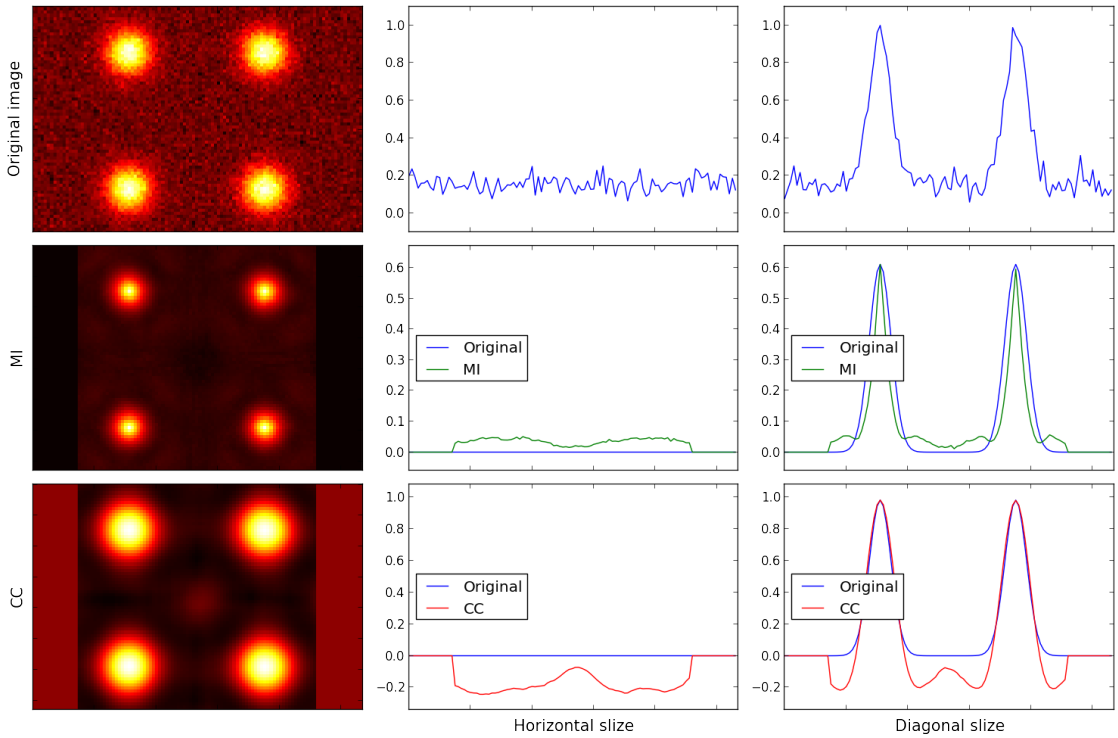

Figure 3.17: Response to an empty space surrounded by 4 particles (upper row) of the MI (middle row) and the CC (lower row) for a minimum peak to peak distance of 30 pixels

In this case we we observe the correct peaks in the diagonal slice for both similarity measures. For the empty center both similarity measures show very little response in the horizontal slice.
Now we lower the minimum distance to 18 pixels.

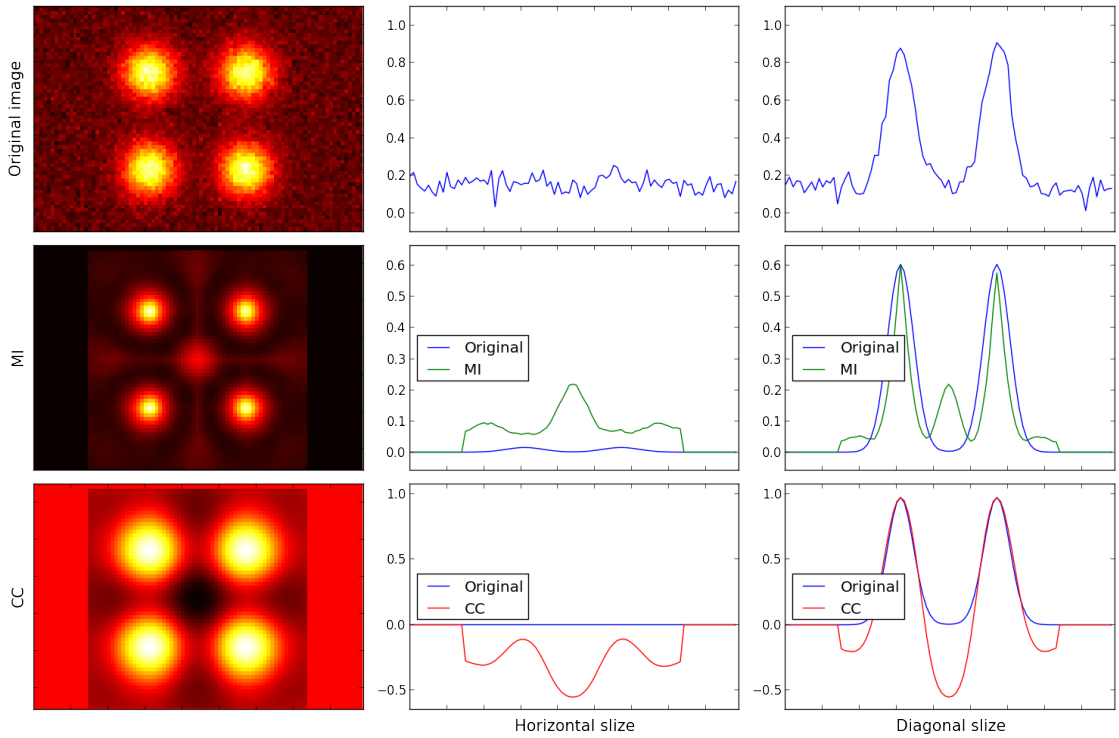

Figure 3.18: Response to an empty space surrounded by 4 particles (upper row) of the MI (middle row) and the CC (lower row) for a minimum peak to peak distance of 18 pixels

While the CC has a minimum at the center of the image, the MI shows a distinct peak in the center of the diagonal slice, as it cannot distinguish between anti-correlation and correlation. In contrast the CC has a minimum, smaller than zero, in the center of the diagonal slice.
The next two figures show the results for a minimum distance of 14 and 12 pixels.

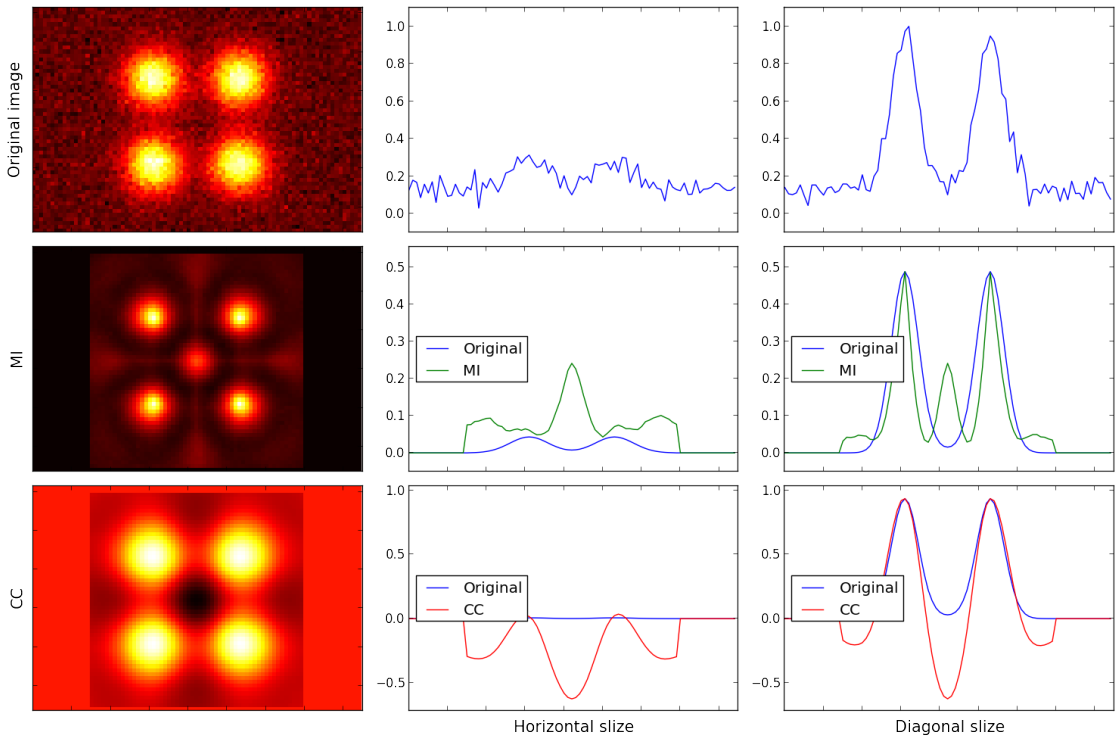

Figure 3.19: Response to an empty space surrounded by 4 particles (upper row) of the MI (middle row) and the CC (lower row) for a minimum peak to peak distance of 14 pixels

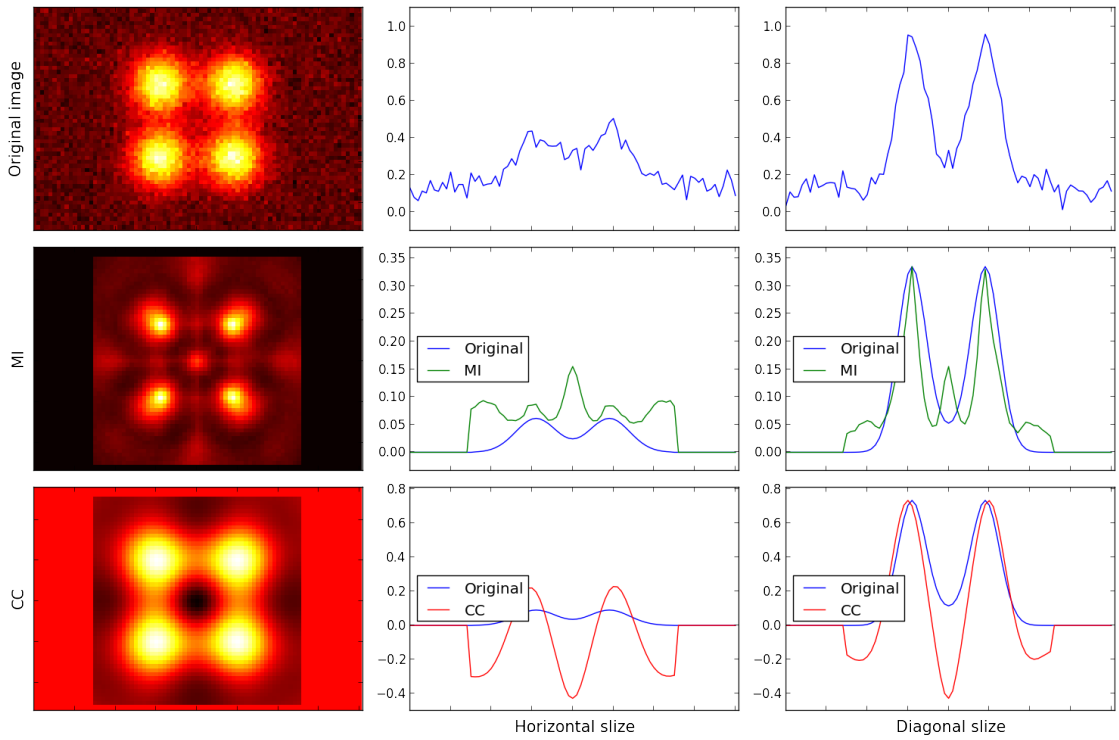

Figure 3.20: Response to an empty space surrounded by 4 particles (upper row) of the MI (middle row) and the CC (lower row) for a minimum peak to peak distance of 12 pixels

In both cases the MI shows a distinct peak in the image center, whereas the CC assigns negative values to the center region.
To summarize, both similarity measures can have their specific problems in high density clouds.
However, the situation where there could be false detections by the MI requires a much more specific constellation of particles as the CC and therefore should be much more unlikely to happen, whereas the situation of only two nearby or even overlapping particle images seems much more probable.
While we see no clear method to overcome the problems of the CC in the case of overlapping particles one could use it to rule out false positive detections of the MI when it comes to false detections due to anti-correlations of empty spaces surrounded by particles.

3.5 Velocity estimation

If particles move with a high velocity they appear elongated in the image. In this case we can use our method to estimate the velocity and the direction of motion of each particle out of one image.
We model a moving particles by an elongated template we introduced in the methods section. If not stated otherwise, all the velocities in this section are given as integer values and have the unit of pixels per frame. These arbitrary unit can be related to a meaningful physical unit by plugging in the resolution, i.e. the width of single pixel, and the exposure time of one frame.

3.5.1 Velocity and direction of motion estimation for one particle

We begin by showing the response of the similarity measures to one particle with different velocities v and $\phi = 0$. In figure 3.21 the original particle is shown in the middle and the response of MI and CC are shown on the left and right, respectively. In this example particle velocities are chosen as zero, five and ten.

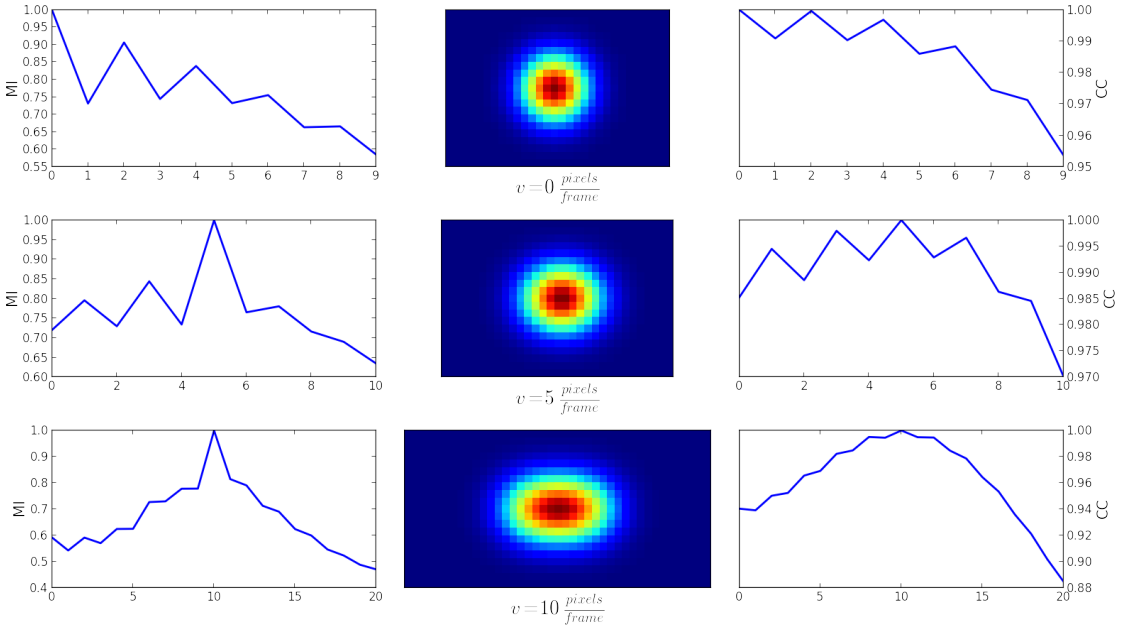

Figure 3.21: Left: MI-response, Right: CC-Response, $SNR = \inf$

As we can see both similarity measures estimate the velocity of the different particles correctly without noise.
Besides the velocity we also estimate the direction of motion, represented by the parameter ϕ, of a particle. The particles all have a velocity of 15 pixels per frame in the horizontal direction and an angle ϕ of $-0.2 \cdot \pi$, $0 \cdot \pi$, and $0.2 \cdot \pi$.

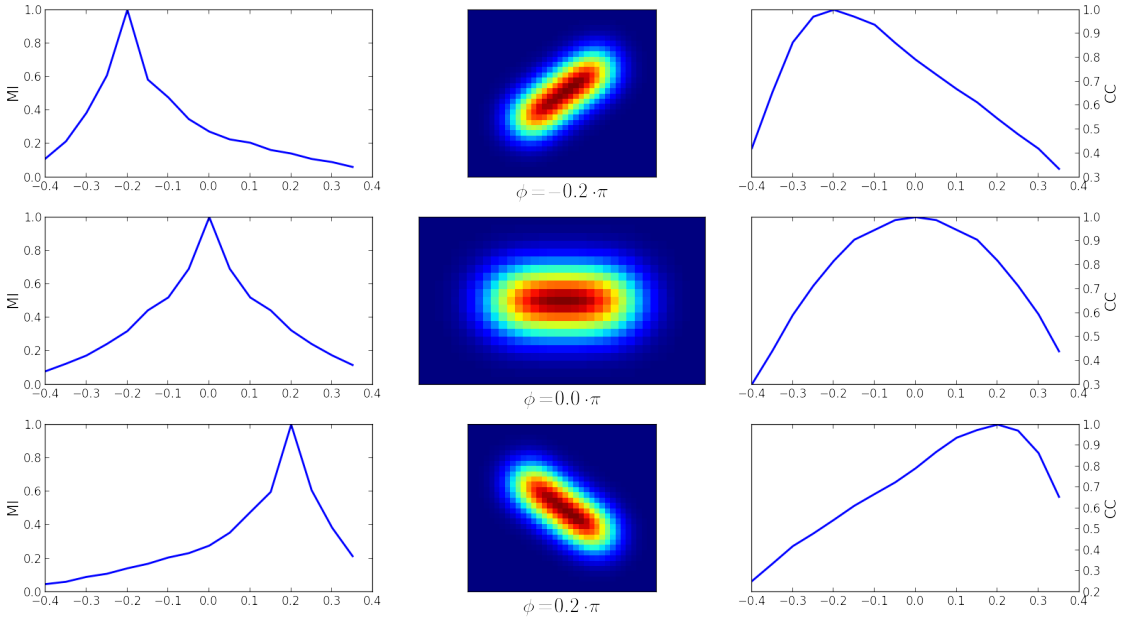

Figure 3.22: Left: MI-response, Right: CC-Response, $SNR = \text{inf}$

As we can see we can both similarity measures show their highest value at the correct velocities and values for ϕ.

3.5.2 Velocity estimation error due to noise and particle size

In this section we demonstrate the stability of the velocity estimation method, which includes the estimation of the absolute value of the velocity and the angle ϕ, under different strengths of noise and for different particle sizes. Here we always assume the most general case of mixed signal dependent and independent noise.
We choose the parameter range for the velocities from 0 to 30 pixels per frame and for ϕ from $-0.4 \cdot \pi$ to $0.4 \cdot \pi$. We test for the velocities and angles independently to be able to examine the differences and similarities in performance for both parameters. For each velocity in the velocity range we create a noisy particle by adding noise to a template with velocity v, while the angle of rotation ϕ is zero for all different velocities. We then calculate the similarity measure value for the noisy particle and every template in the velocity range from 0 to 30 pixels per frame. The estimated velocity corresponds to the maximum value of the similarity measure with respect to the template velocity. We perform this calculation 100 times and calculate the mean value and standard deviation for the 100 velocity estimations. To visualize the results we plot the estimated mean velocity values against the true velocity and indicate the error by the standard deviation.
For the estimation of the angle ϕ we perform the same calculations. Only here we keep the velocity at a constant value of 15 pixels per frame in x-direction and vary ϕ in steps of $0.01 \cdot \pi$ from $-0.4 \cdot \pi$ to $0.4 \cdot \pi$.

One thing has to be kept in mind when it comes to the method for the velocity estimation and the estimation of ϕ. The procedure of estimating these parameters out of a single image is based on the imaged shapes of the particles. Therefore one expects the size of the particle to influence the accuracy of the velocity estimation algorithm. We therefore repeat all of the calculations for three different particle sizes of one, three and five pixels standard deviation.
In figure 3.23 we show the results for the estimation the velocity and in figure 3.24 the estimation of ϕ for a particle size of standard deviation one and different SNR values.

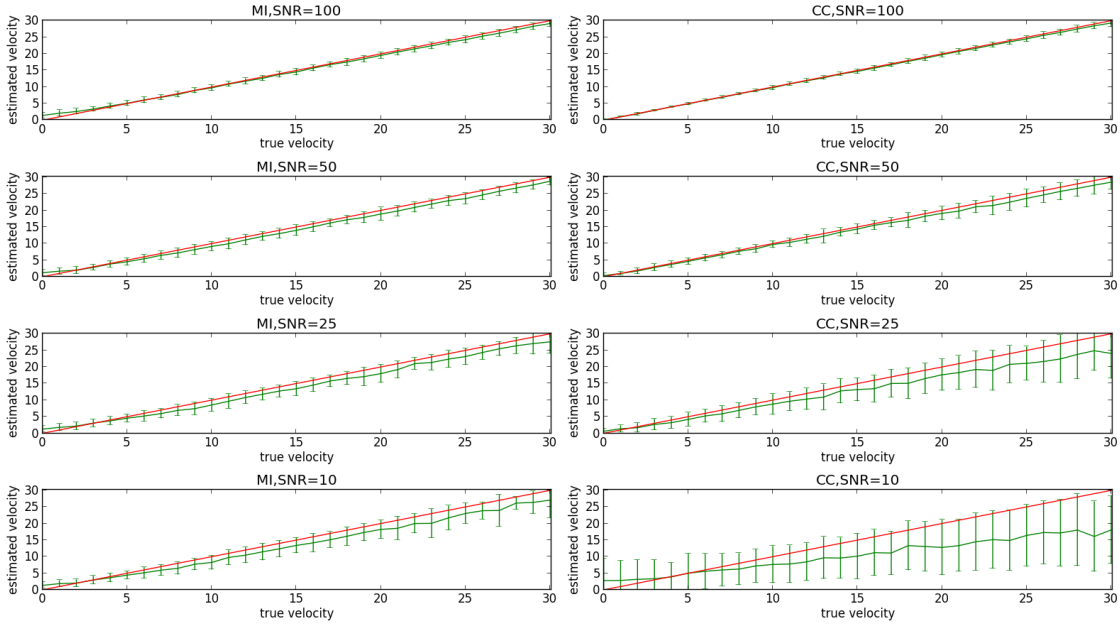

Figure 3.23: Velocity estimation for particles with one pixel standard deviation

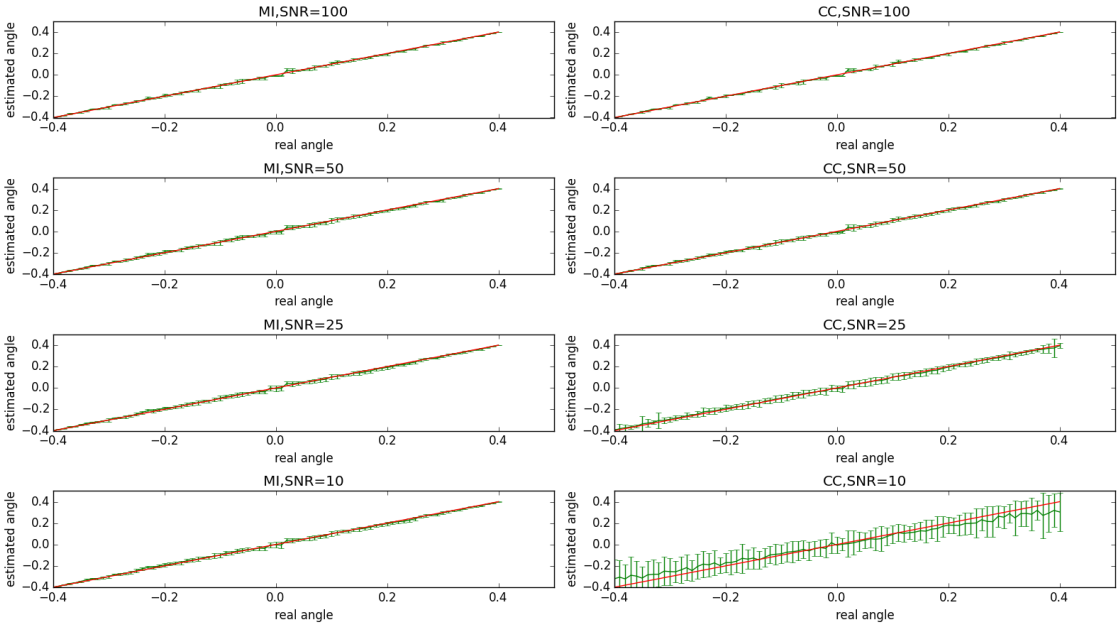

Figure 3.24: Estimation of ϕ for particles with one pixel standard deviation

For this particle size the estimation of both parameters is much more accurate in the case of the MI. We observe a general trend in the velocity estimation to underestimate higher velocities for increasing noise. While this effect is rather moderate for the MI, for the CC it is much stronger.
Increasing the particle size to a standard deviation of three pixels we show the performance of the similarity measures in figures 3.25 and 3.26 .

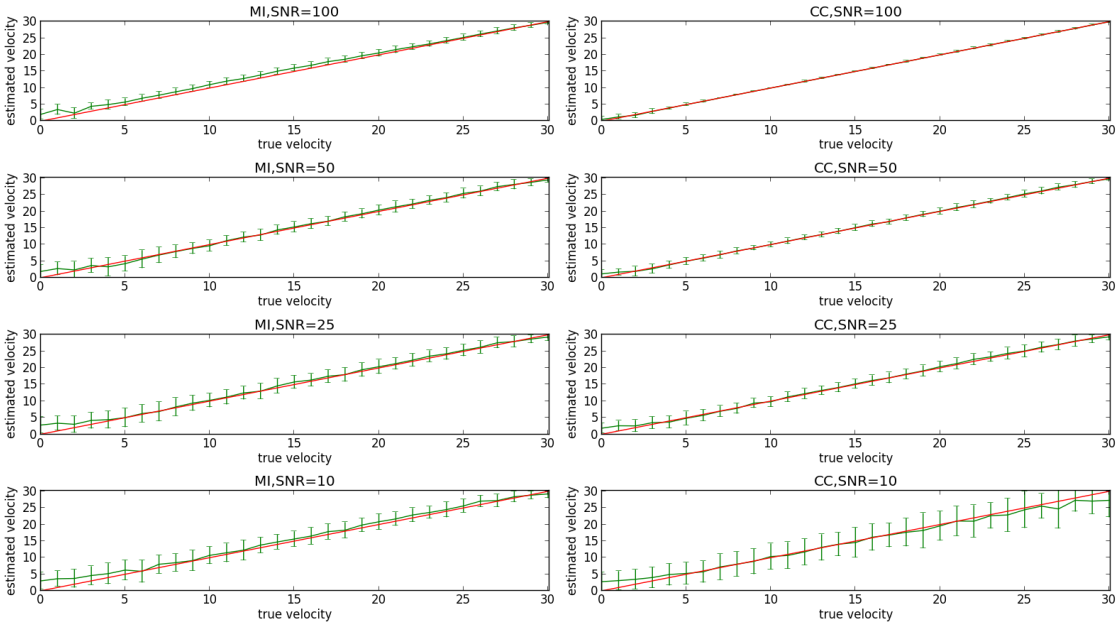

Figure 3.25: Velocity estimation for particles with three pixels standard deviation

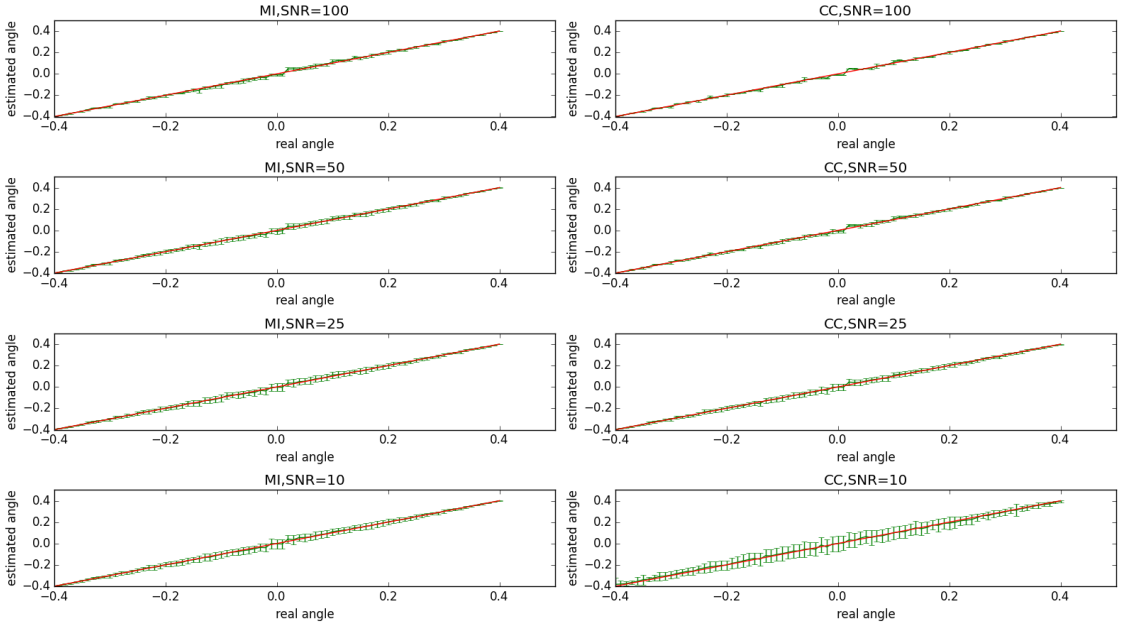

Figure 3.26: Estimation of ϕ for particles with one pixel standard deviation

In case of the MI the results are nearly unaffected for increasing noise, except now low velocities tend to get overestimated. On the contrary, the CC performs much better than for the smaller particles. The error of the angle estimation is slightly increasing for angles close to 0. Again as for the particles of size one standard deviation this effect is stronger in the case of the CC than in the case of the MI.

Now we again increase the particle size to five pixels standard deviation and obtain following results.

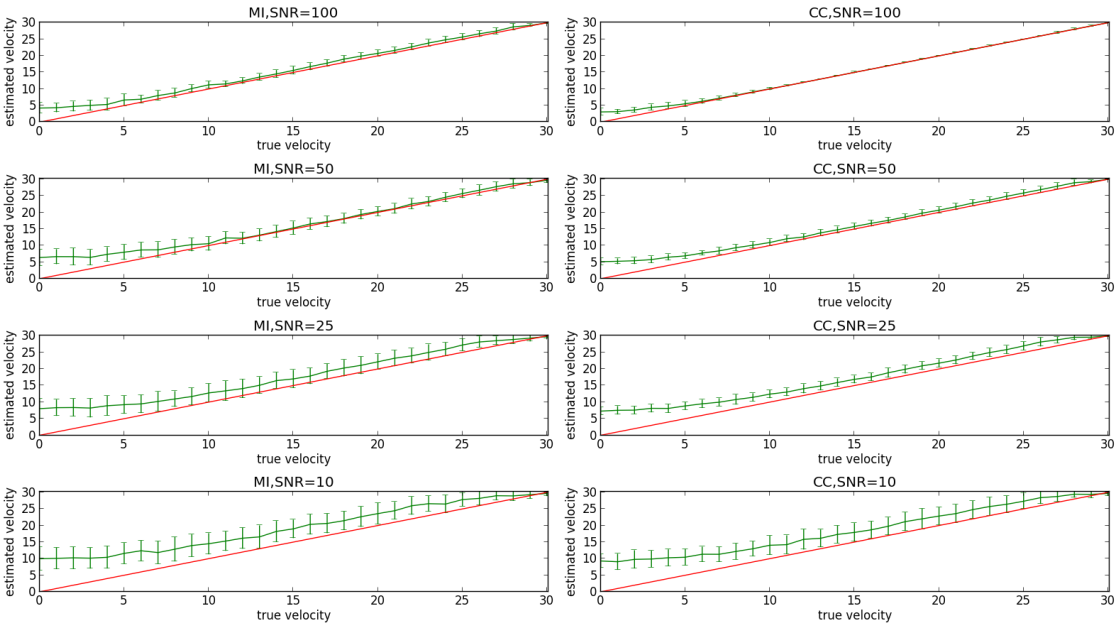

Figure 3.27: Velocity estimation for particles with five pixels standard deviation

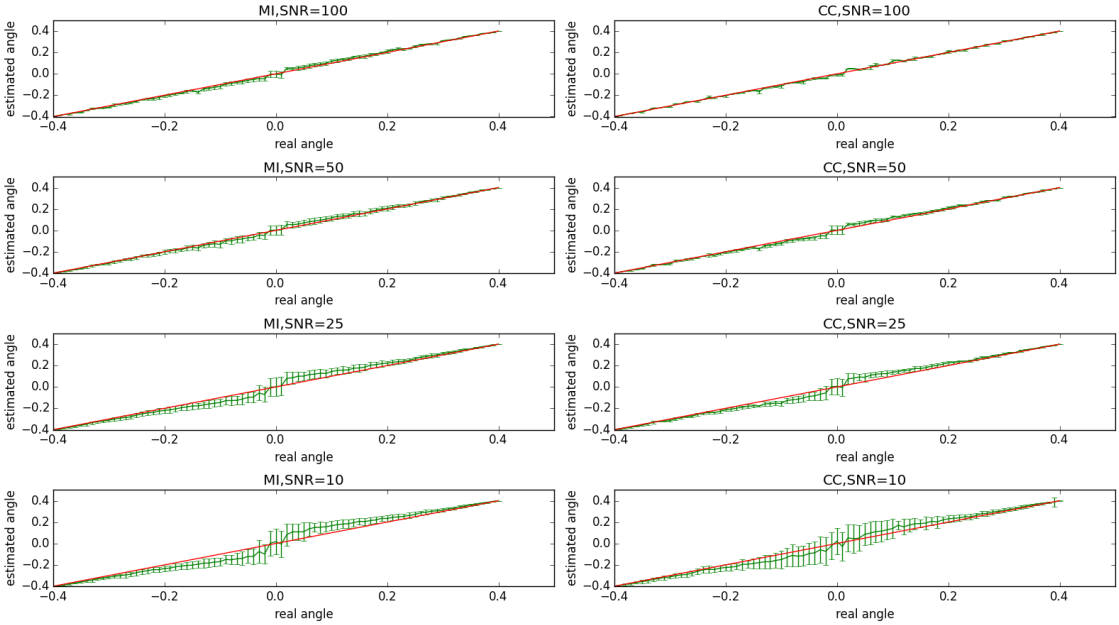

Figure 3.28: Estimation of ϕ for particles with one pixel standard deviation

The trend of overestimating slow particle velocities grows stronger for both similarity measures. Also in the angle estimation we observe an increase in the error for

small angles compared to the particle size of three pixels standard deviation.
We observe that in general for slow velocities, ten pixels per frame and below, the velocities tend to get overestimated. This effect is quite moderate for one and three pixel particle sizes but very strong for five pixel particle sizes. For the higher velocity range MI has the better performance for one and three pixel particle sizes. For a particle size of standard deviation of 5 both similarity measures show worse performance. Similarly the error in the angle estimation increases with increasing particle size and noise.

3.5.3 Velocity profile estimation

One of the typical application of our velocity estimation algorithm is the estimation of the velocity profile of a fluid. While the velocity of the fluid cannot be measured directly with our method, it could be achieved by imaging tracer particles in the fluid.
We model such a fluid, containing tracer particles, where we distribute particles in a image, with velocities according to a velocity profile we choose. One obtains these velocity profiles by estimating the velocity of the flow as a function of the position in the system over time and then by averaging the estimated velocities over time.
We use velocity profiles of laminar and turbulent pipe flows to create images containing particles with velocities according to these profiles. Our goal is to recover the known profiles in the presence of image noise. In reality not only the images suffer from noise, but also the velocities do not follow the velocity profile but fluctuate around it. Therefore, to create a more realistic situation we not only add noise to the final image containing the particles, but also to the velocity profile itself that we use to obtain the particle velocities. For a detailed discussion on the velocity profiles we use please refer to [47].
In the laminar case the velocity, as a function of the displacement from the center of the pipe, can be described by the following function

$$v(r) = v_{max}(1 - \frac{r^2}{R^2}) \tag{3.6}$$

$$r = |y - R| \tag{3.7}$$

Here v_{max} denotes the maximum velocity of the flow and R the the radius of the pipe. To simulate the fluctuation around the velocity profile we add noise to the profile according to a zero mean Gaussian distribution with a standard deviation of one.

$$v_{noise}(r) = v(r) + \mathcal{N}(0, 1) \tag{3.8}$$

Especially for low velocities in the uncorrupted velocity profile there is a chance for negative values in the noisy velocity profile due to the range of the Gaussian distribution. We require all particles to flow in the same direction, because our method cannot distinguish a particle moving forward and backward. Distinguishing a particle moving forward from a particle moving backwards is only possible if one

uses at least two frames and tracks a particle in these two frames. As we want to show the estimation of the velocity profile, without any additional tracking algorithm, we set all negative velocity values of the noisy velocity profile to zero

$$v_{noise}(r) = \begin{cases} v_{noise}(r), & \text{if } v_{noise}(r) > 0 \\ 0, & \text{else} \end{cases} \tag{3.9}$$

In figure 3.29 we show an example image containing 50 particles with velocities according to a noisy laminar velocity profile. One can see that the velocities in the image fluctuate, but generally follow the parabolic shaped velocity profile we described.

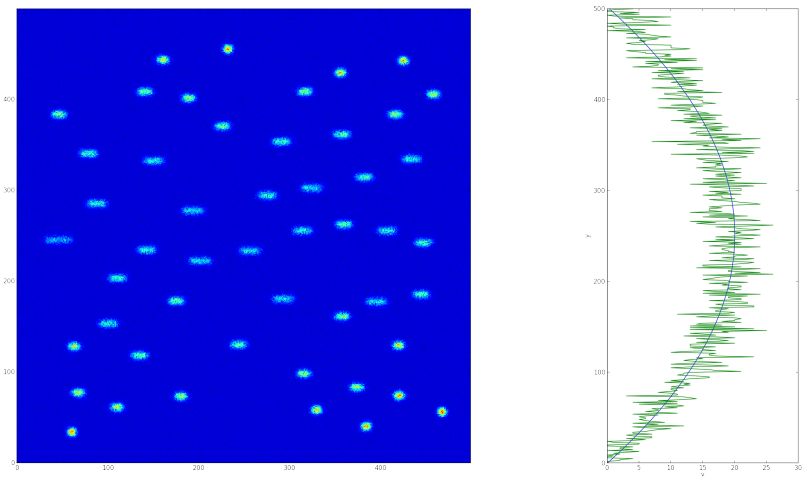

Figure 3.29: Example image containing 50 particles (left) with velocities according to a noisy laminar velocity profile (right)

In the case of the turbulent pipe flow one cannot express the velocity as a function of the distance to the pipe center any more. However the velocity profile follows the following functional form pretty well

$$v(r) = v_{max}(1 - \frac{r}{R})^{1/n} \tag{3.10}$$

$$r = |y - R| \tag{3.11}$$

where $n = 7$ suits a wide range of cases and is commonly referred to as the one seventh power law. Again v_{max} corresponds to the maximum velocity of the particles and R to the distance of the center to the edge of the pipe. Whereas in the laminar case the velocities in the image follow 3.6, with small deviations, for all time steps, this assumption cannot be made in the turbulent case. We simulate this fact by increasing the noise on the velocity profile by a factor of three, creating a much

higher variety of velocities over the whole profile.

$$v_{noise}(r) = v(r) + \mathcal{N}(0, 3) \tag{3.12}$$

As for the laminar velocity profile we set all negative velocities, resulting from the noise we apply to the velocity profile, to zero.

$$v_{noise}(r) = \begin{cases} v_{noise}(r), & \text{if } v_{noise}(r) > 0 \\ 0, & \text{else} \end{cases} \tag{3.13}$$

An example of an image with 50 particles with a velocity distribution following such a noisy turbulent velocity profile is shown in the figure 3.30.

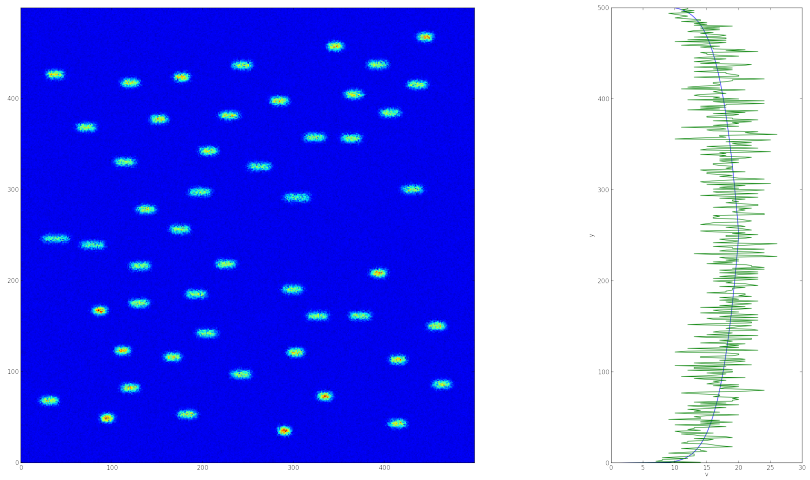

Figure 3.30: Example image containing 50 particles (left) with velocities according to a noisy turbulent velocity profile (right)

For recovering the initial velocity profile we use 30 images containing 50 particles each with velocities according to the velocity profile under consideration. The noise values for the velocity profiles and the images itself are drawn independently for every image. Also the particle positions in the different images are independent of the particle positions of the other images. This assumption is valid for particles moving in the same direction because we can estimate the velocities of the particles out of one frame, the information of the position of the particle in the previous and subsequent frame is not required. On can think of the 30 images we create as 30 snapshots of the time evolution of a system with a large time delay in between.
After we estimated the velocities for all of the particles of the 30 images, as a function of the y axis, we calculate the mean velocity value, again as a function of the y axis. As a last step we fit these average velocity values to recover the original velocity

profile.
The fitting functions for the laminar and turbulent regimes are

$$\begin{aligned} \text{laminar regime: } v(r) &= a(1 - \frac{r^2}{b^2}) \\ \text{turbulent regime: } v(r) &= a(1 - \frac{r}{b})^{\frac{1}{c}} \end{aligned} \tag{3.14}$$

with fitting parameters a, b and c. We repeat this process for the laminar and turbulent cases for different SNR values of 100, 50 and 25.
In figure 3.31 we show the results for the laminar velocity profile with a SNR value of 100.

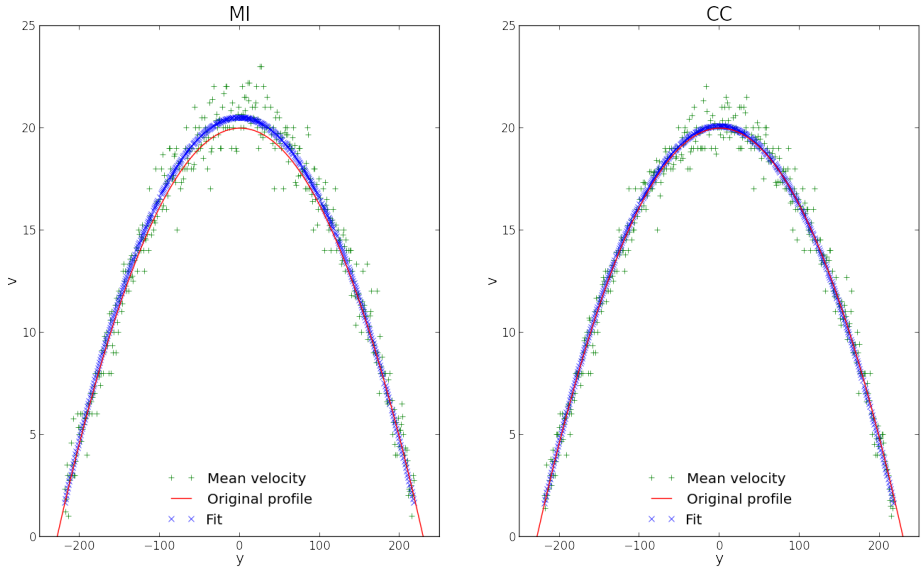

Figure 3.31: Velocity profile estimation of laminar flow profile with a SNR of 100

Here we can see that we can nearly perfectly recover the original velocity profile in case of the CC. In the case of the MI we can observe a slight discrepancy between the fitted velocity profile and the original velocity profile in the center of the flow. Outside the center region both similarity measures estimate the velocity profile very accurately.
The figure 3.32 shows the results for a SNR value of 50.

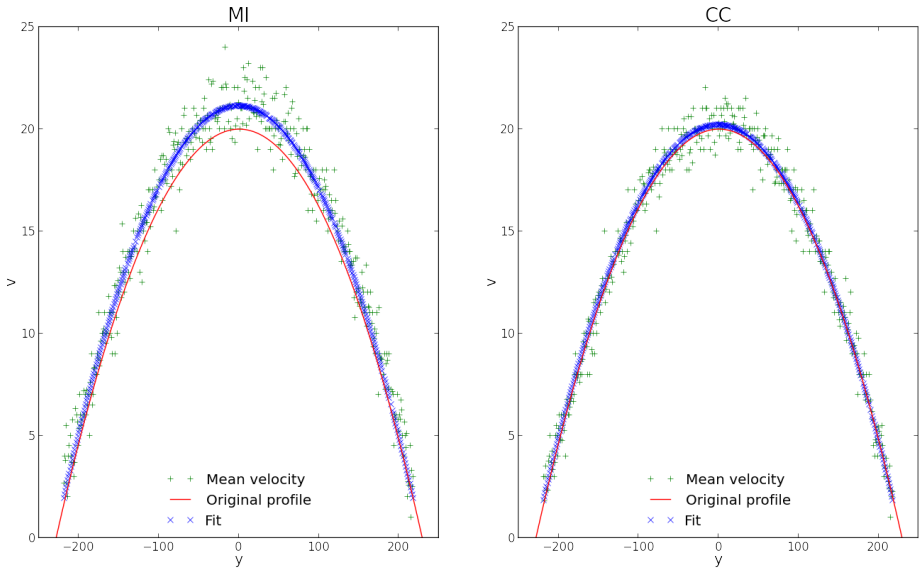

Figure 3.32: Velocity profile estimation of laminar flow profile with a SNR of 50

While the CC stays nearly unaffected, the trend for overestimating the velocity in the center of the flow in case of the MI grows stronger. Still on the edges both similarity measures perform very well.
For a SNR value of 25 the results are shown in figure 3.33.

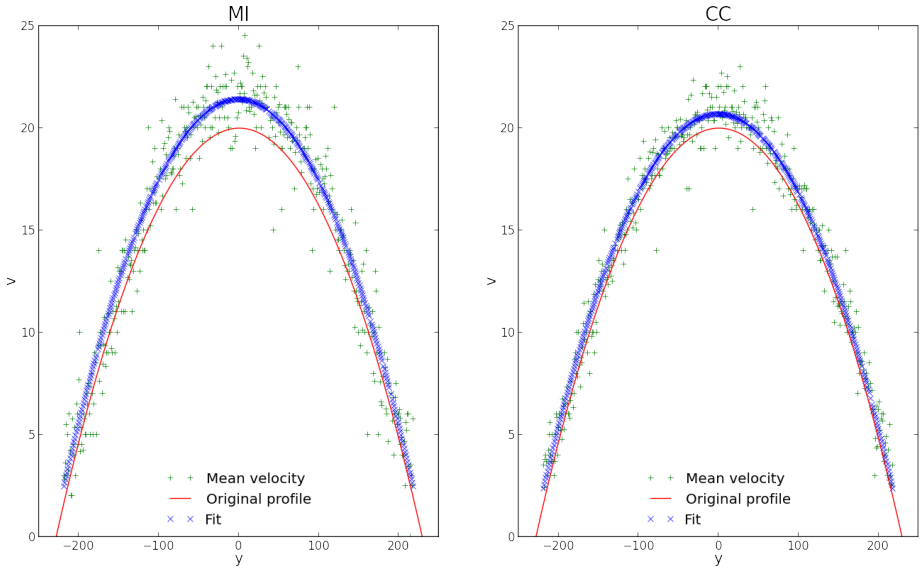

Figure 3.33: Velocity profile estimation of laminar flow profile with a SNR of 25

Here the effect of overestimating the velocities in the center of the flow are observable for both similarity measures. However the CC is still more accurate overall for this noise level.
In the next three figures 3.34, 3.35 and 3.36 we present the results in the turbulent regime for SNR values of 100, 50 and 25. Here we expect a larger deviation from the true velocity profile, as we have increased the noise on the original velocity profile by a factor of three. An additional complication is that here we have to include another fitting parameter c, which in general complicates the fitting of data.
For a SNR value of 100 we present the results in figure 3.34.

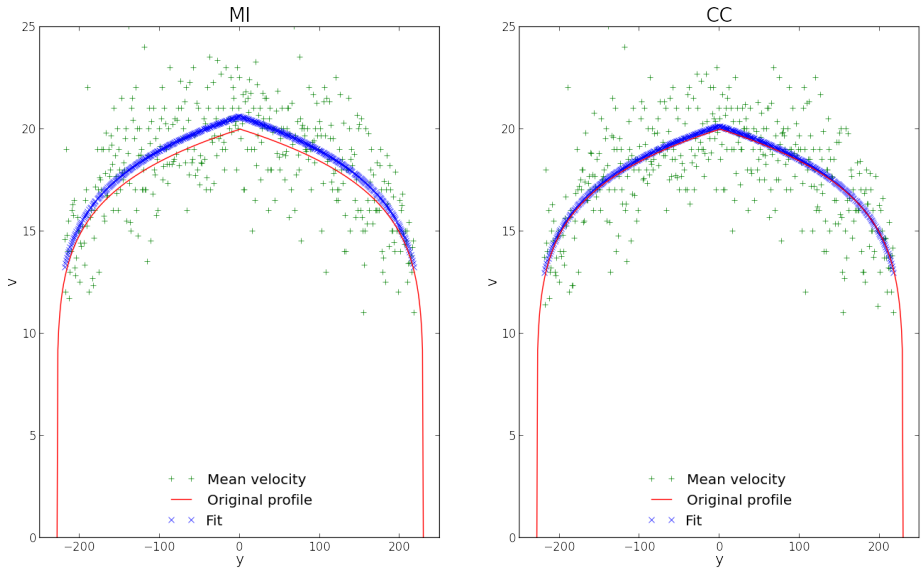

Figure 3.34: Velocity profile estimation of turbulent flow profile with a SNR of 100

Again the MI overestimates the center of the velocity profile like in the laminar regime. On the contrary the CC fits the velocity profile in the center very precisely but is slightly more imprecise on the flanks.
In the case of a SNR value of 50 the results are presented in figure 3.35.

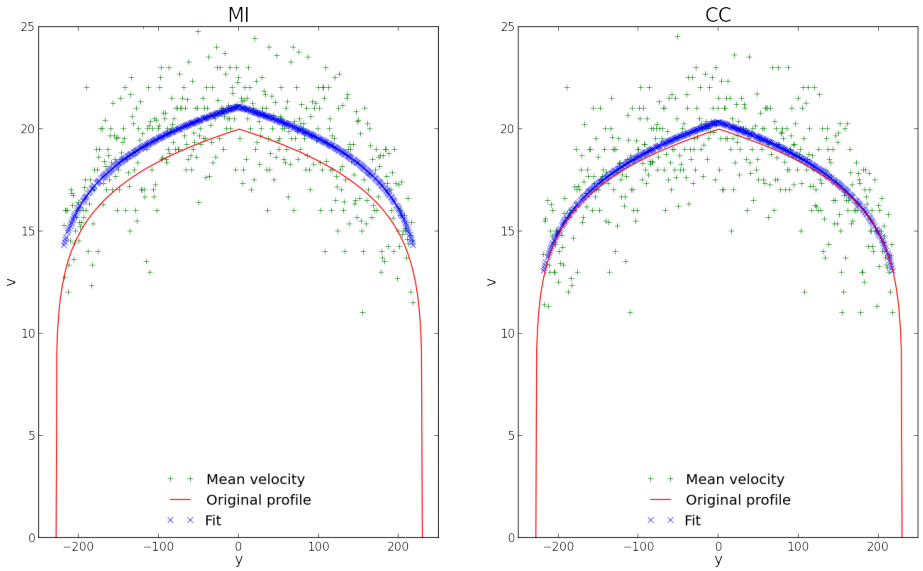

Figure 3.35: Velocity profile estimation of turbulent flow profile with a SNR of 50

Interestingly the MI estimates all the values to high, but seems to follow the general trend of the profile better than in the previous case. Also, the CC in general estimates the mean velocities higher than in the previous figure. And finally for a SNR value of 25 the results are shown in figure 3.36

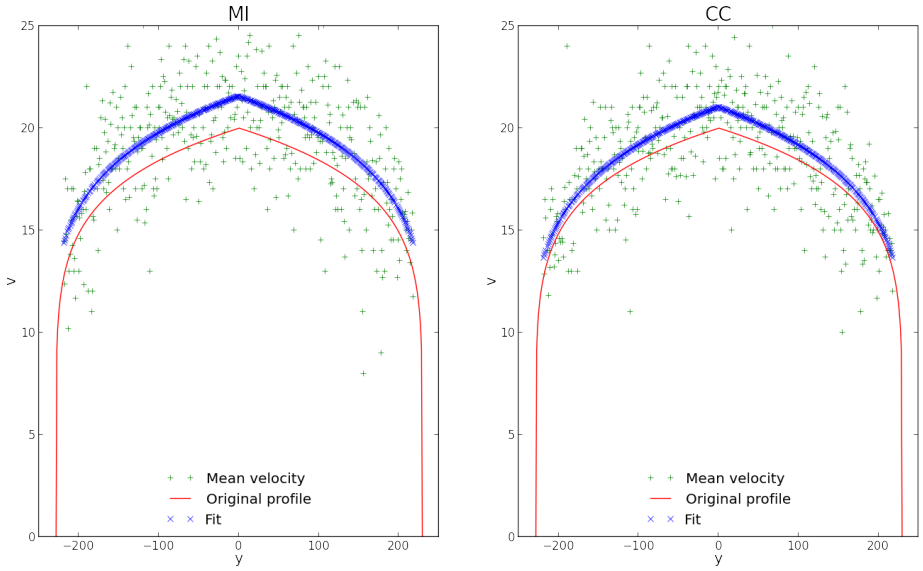

Figure 3.36: Velocity profile estimation of turbulent flow profile with a SNR of 25

Here, the original profile is not recovered very well by either the MI or the CC. In this section we showed that we can recover velocity profiles with our method for both the MI and the CC for low noise values. Overall the CC performs better in terms of accuracy and stability due to noise than the MI. As we expected, the estimation process in the turbulent regime is more complicated and both similarity measures suffered from the additional complications, namely the higher noise on the profile and the more complicated fitting function. As a possible explanation for the worse behavior in this section compared to for example the pure position estimation, in terms of stability due to image noise, we propose following argument. In the pure position estimation the particles in the uncorrupted images were all of equal brightness and therefore all equally affected by the noise in the image. In this section we have wide variability of particle velocities and therefore a wide variability in brightness of the particles. As the SNR value is defined over the maximum intensity in the image the fast particles are much more affected by the noise than the slow ones. Evidence for this can be found in the laminar case, where the biggest error occurs for the fastest particles in the middle. In the turbulent regime the velocity profile decreases much slower than the laminar profile. Therefore we have much more particles with larger velocities in this case which should be affected by noise more than the slow particles.
Nevertheless if the noise in the images is not to high, we can recover the original profiles very well in both, the laminar and the turbulent, flow regimes.

Chapter 4

Experimental data

In this section we apply our methods to experimental data from the PK-4 experiment which is currently located on the Columbus module on the International Space Station (ISS). The PK-4 experiment is designed for the investigation of a complex plasma in a dc discharge.
The plasma is produced by a voltage of about 1000 V applied to the dc electrodes in a neon or argon gas discharge at pressures of 10 to 200 Pa. The microparticles, made of melamine formaldehyde and of diameters between 0.5μm and 11μm, are suspended in the plasma. There they acquire a negative charge of the order of several thousands of elementary charges due to electrons streaming onto their surface. The particles then interact through the electric force, and can form regular structures, behave as liquids or exhibit collective phenomena such as waves. For a detailed presentation of the experimental setup please refer to [48],[49],[50].
The kinematics of the complex plasma are recorded by the scattering light of the microparticles illuminated by a laser and saved as gray value images. As the laser illuminates only a thin sheet of the whole three dimensional dust cloud, the images always show only a thin slice of the cloud. A typical image containing a slice of the dust particle cloud is shown in the figure 4.1.

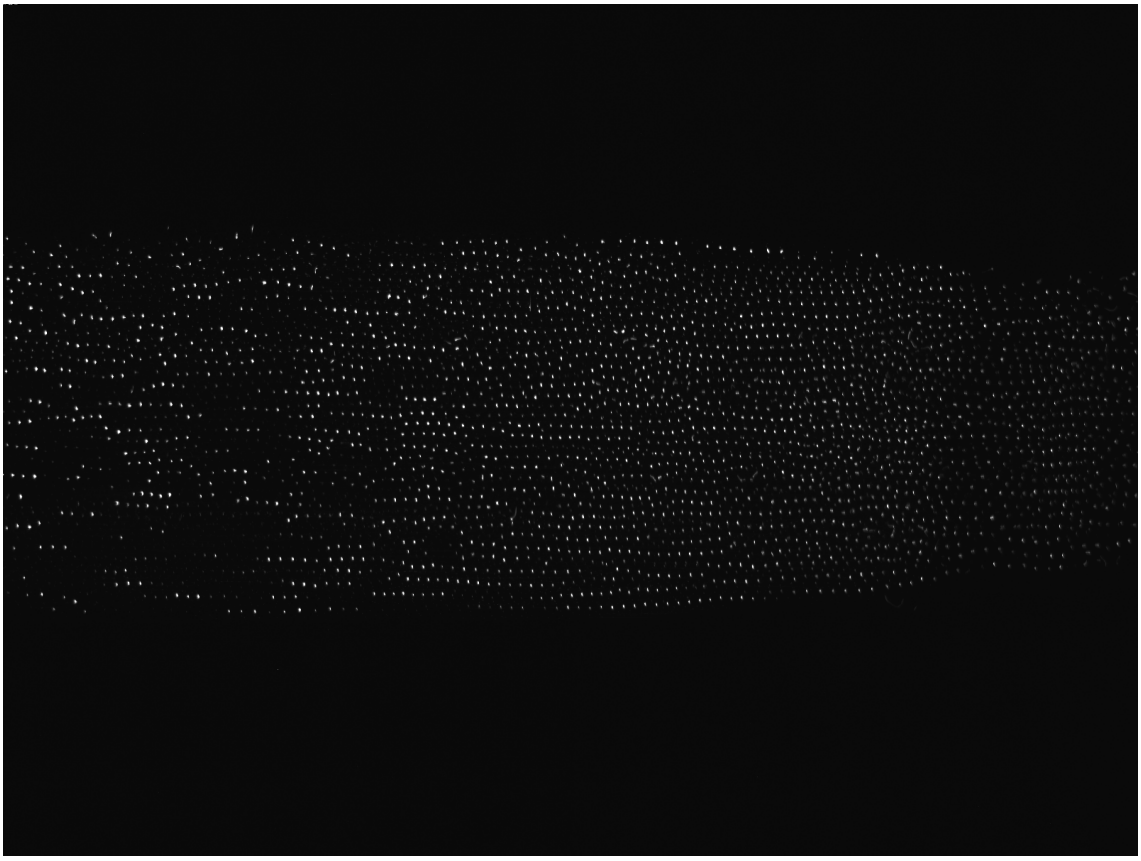

Figure 4.1: Example image of a slice through the dust particle cloud of the PK-4 experiment in microgravity

Besides from the illumination laser, the PK-4 setup involves also a manipulation laser with which certain regions of the particle cloud can be accelerated. The manipulation laser creates a velocity gradient perpendicular to the laser. An example of such a laser driven velocity gradient is shown in figure 4.2.

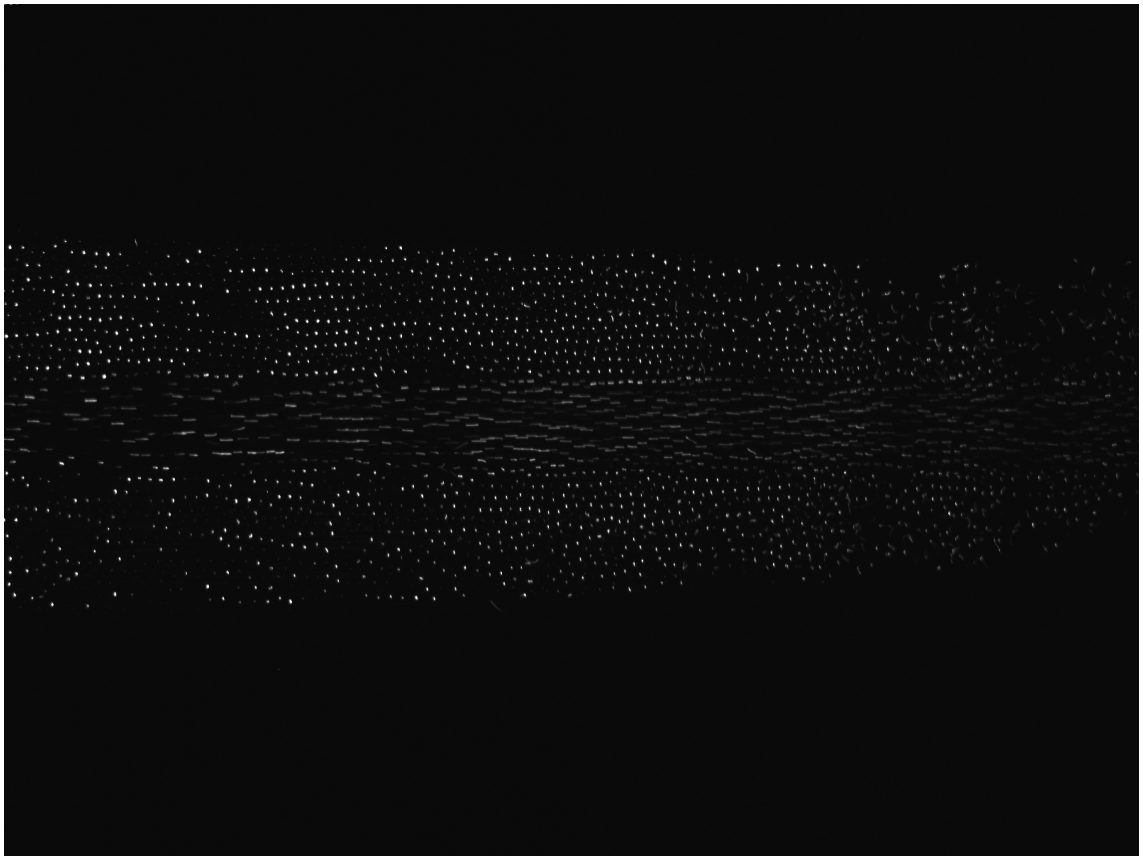

Figure 4.2: Example image of a dust particle cloud accelerated by a six watt laser. The laser is centered at the vertical center of the image.

Here we can clearly see the elongated particles in the center of the image accelerated by the laser. As described before, we can use the elongation of the particles to estimate their velocities. We assured before that the main direction of motion for all of the particles is in the same direction, such that we can estimate the velocities out of a single image, without the need to track the particles.
Even though the data we are presenting here were originally for commissioning purposes, i.e. beyond optimal conditions, we still apply our methods. Nevertheless one should keep in mind, that the data is not taken under the optimal conditions and we therefore expect these result to improve for further optimized data.

4.1 Position estimation of dust particles

To compare results in position estimation for experimental data is naturally rather less straightforward than in the case of synthetic data, where the ground truth about position and size of the particles is known. However, we can compare results of both similarity measures by eye as most of the particles are clearly visible in the image if they are in the middle of the laser sheet. Particles which are at the edges of the

laser sheet appear much darker in the image and are almost invisible for the eye. Nevertheless, these dark particles get detected by the particle detection method. To make them visible we show the images on a logarithmic scale, reducing the dynamic range of the brightness values in the image. We can therefore more easily judge the performance of the different similarity measures also for the particles at the edges of the laser sheet.
Here we show some examples for particles with low peak to peak distances in an image. In figure 4.3 and 4.4 we show the particle detection performance for both similarity measures for experimental data from the PK-4 experiment where several particles are close together. On the top row of the figures we show in the middle the original image with the detected particle centers, on the left by the CC and on the right by MI, denoted as red dots. On the lower row we show the in the middle the corresponding similarity measure maps we used to detect the particles. Both maps were calculated with an isotropic template with standard deviation of one pixel and a window size of 11 pixels. The threshold for the maximum detection is 30 percent of the maximum value in the respective similarity measuer maps and was chosen by hand to maximize the performance of both similarity measures. Due to the large image sizes of the experimental data we show subsections of the images for demonstration purposes. Besides the images we show zoomed in sections, denoted by red squares in the images and similarity measure maps.

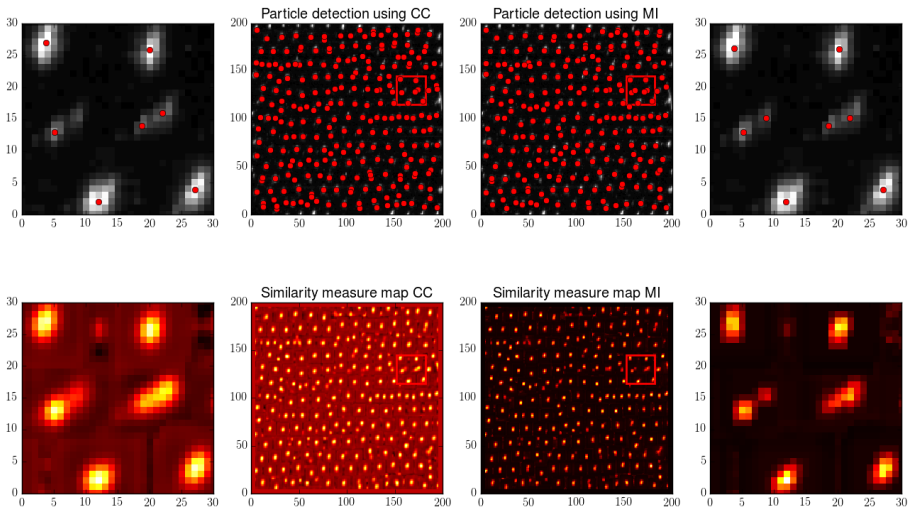

Figure 4.3: Position estimation of dust particles in the PK4 complex plasma experiment

In the middle of the zoomed in sections we observe two pairs of two nearby particles each. While both particles of the pair on the right are detected by both similarity measures correctly, the pair on the left is only detected correctly by the MI. If we look at the corresponding sections of the similarity measures we can observe

that for the left pair the response of the CC is a highly bright spot for the left particle and rather dark for the right one. In the case of the MI we observe two distinct spots of much more alike brightness. This probably causes the MI to separate the two particles while the CC only detects the left particle.
Another example for the differences in particle detection of the two similarity measures is shown in figure 4.4.

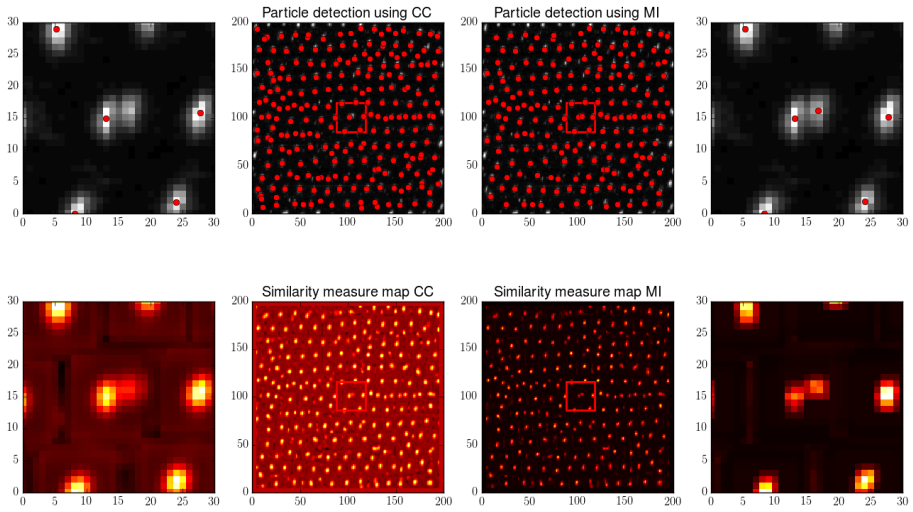

Figure 4.4: Position estimation of dust particles in the PK-4 complex plasma experiment

Here we show two particles close to each other in the zoomed in sections. Again the MI detects two particles where the the CC only detects one particle. Here we observe the effect for differences in brightness for two nearby particles for the CC even stronger than in the example before. In the case of the MI, the response to the two particles is practically of equal brightness.
In summary, in the case of the CC one can observe that if two particles are close together, one of the spots in the similarity map can be significantly darker which can cause the maxima detection algorithm to disregard it because it falls beneath the threshold. In contrast the spots of two close particles in the similarity map calculated by MI are usually more similar in brightness causing the algorithm to detect both particles. As we noted in the beginning of this section one can only judge by eye the correctness of the results. By this results we see strong evidence for the advantage of the MI over the CC in particle detection in case of dense particle clouds which we already showed for synthetic data in section 3.4.

4.2 Velocity estimation of dust particles

To take our investigation of the experimental data one step further we apply our velocity estimation method to images of the PK-4 experiment where the particles in the image center are accelerated by a laser along the horizontal axis of the images, resulting in a velocity gradient along the vertical axis. To estimate the velocities of the particles in the image we use an isotropic template of standard deviation of 1 pixel from which we build a series of elongated templates with velocities ranging from 0 to 30 pixels per frame.
To express the estimated velocities in pixels per frame in physical units we use the inverse of the frame rate of $35Hz$, which is the exposure time in this case, and the resolution of $14,2\mu m$ per pixel of the camera.

$$1\frac{\text{pixel}}{\text{frame}} \widehat{=} 0.00049\frac{m}{s} \tag{4.1}$$

In the following figures we give an example the combined estimation of position and velocity of the particles in a single experimental image estimated by MI in 4.5 and by CC in 4.6. The particles are accelerated by a laser of approximately $6W$.
On the left we plot the original image with the estimated particle position plotted as coloured dots. The color coding of the dots resembles the velocity of the respective particle. On the right we plot the estimated velocities against their corresponding y-axis value.

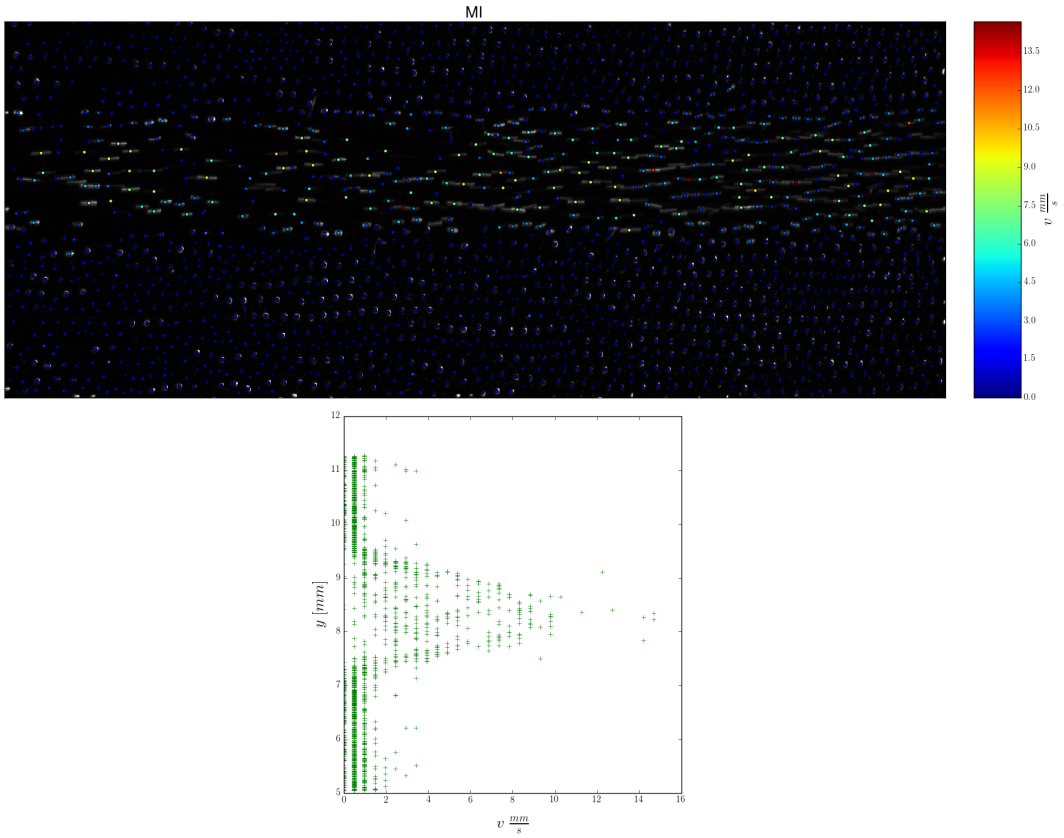

Figure 4.5: Laser driven particle flow: original image with estimated particle positions by MI (color coded according to the estimated velocity) on the top and the y-position of each particle plotted against its velocity on the bottom

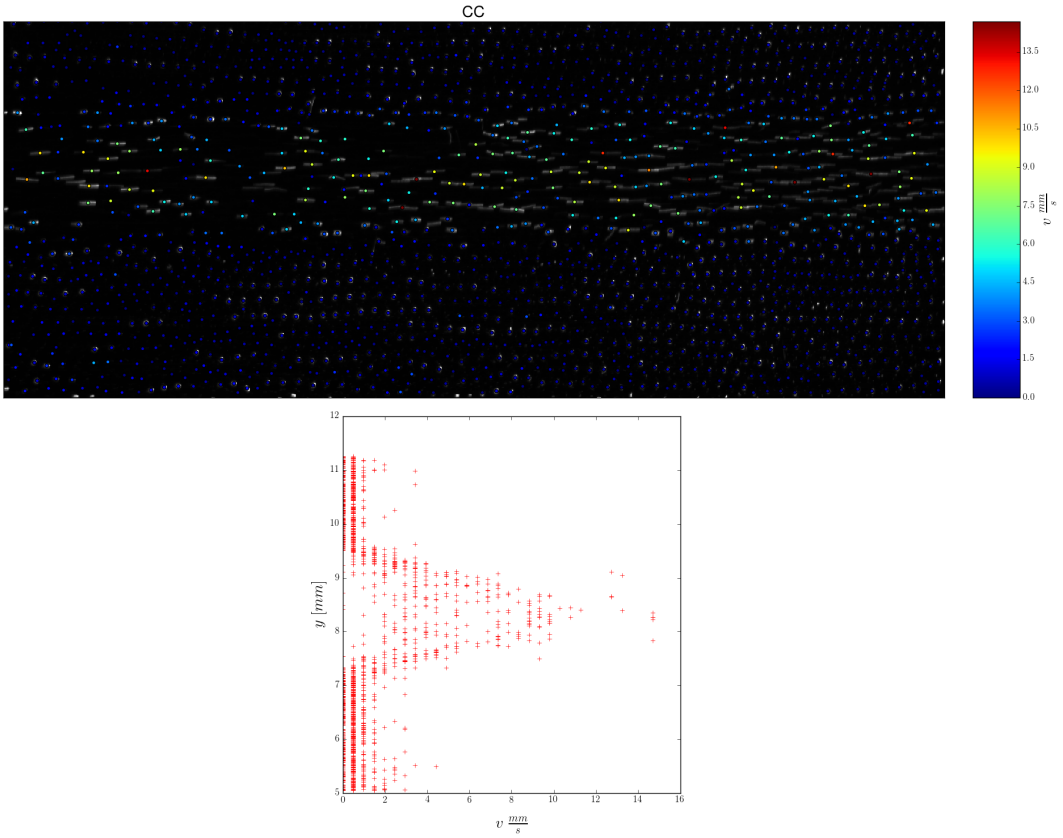

Figure 4.6: Laser driven particle flow: original image with estimated particle positions by CC (color coded according to the estimated velocity) on the top and the y-position of each particle plotted against its velocity on the bottom

We can observe the impact of the laser on the estimated particle velocities very clearly. The center of the image, which corresponds approximately to the center of the laser, shows the highest velocities. With increasing distance from the center of the laser the velocities decrease accordingly.

4.3 Velocity profile estimation

The velocity profile of a fluid or fluid like system can be used to estimate physical quantities such as the viscosity of a fluid. We apply our method to estimate the particle velocities over time in order to calculate the time averaged velocity profile of the complex plasma. To achieve this, we estimate the velocities over 150 frames, with a constant laser power of $6W$.
One assumption for a meaningful velocity profile is that the system is in a steady state, and the velocities fluctuate around a constant mean value over time. To check if the system is in a steady state we calculate the mean velocity in the center of the image and fit a linear function of the form

$$f(v) = m \times v + c \tag{4.2}$$

to assure the steady state. The parameter m, the slope, should ideally be zero, if the system is in a steady state. The offset c represents the constant mean value over time. Because the first frames we investigate are taken before the laser is turned on, we disregard the first twenty values for the fitting process. In figure 4.7 we show the time evolution of the mean velocity of the particles in a box in the image centered around the laser.

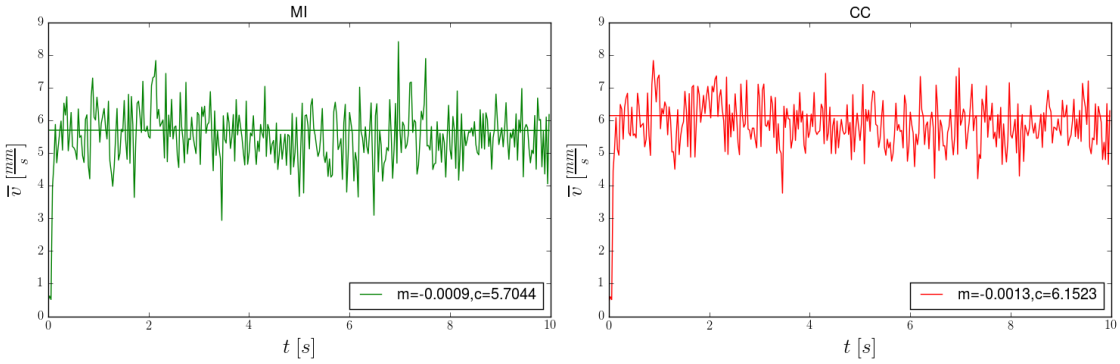

Figure 4.7: Time evolution of the mean velocity in a window centered around the laser beam for a laser power of 6 W

The fit of the time evolution is nearly constant over time. In the time evolution of the mean velocity we observe a very strong acceleration of the particles at the beginning. In just about two to three frames, after the laser was turned on, the mean velocity has reached a value near the constant mean value of the fit c. The

slopes of the fits are both in the order of at least 10^{-3}. This justifies the calculation of a time averaged velocity profile.
In figure 4.8 we show the corresponding velocity profile, calculated in a section of 550 pixels height and 800 pixels width. The section starts at the left edge of the images and is centered by eye around the center of the accelerated region in the image.

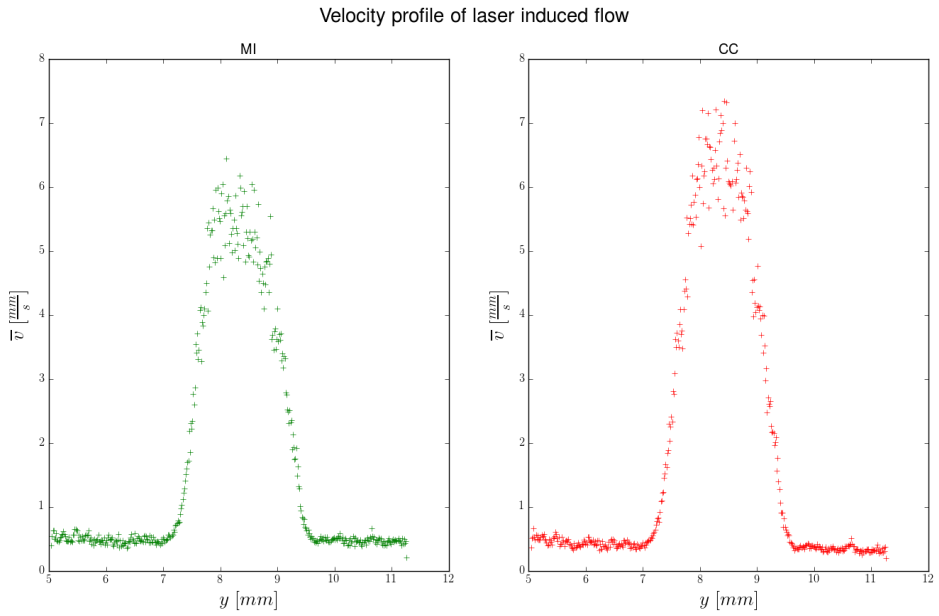

Figure 4.8: Velocity profile for laser driven particle flow in the PK4 experiment accelerated by a 6 W laser.

We can see that the estimated velocity profiles are fluctuating around a small value outside the center region. Above about 7.2 mm and below 9.5 mm we observe an increase in velocity for both similarity measures. After that follows a steep, narrow increase in the mean velocity with a strong fluctuating region in the middle of the profile. In general, the velocity estimation according to the CC results in higher values.
As we do not know the exact position and width of the laser we can only assume that this strongly fluctuating region corresponds to the width of the laser. If that is the case we can identify the narrow regions of the velocity profile outside the fluctuating center as the region where the flow is induced by shear forces. This part could therefore be used to estimate the viscosity of the system. Evidence for this assumption gives the previously measured beam profile of the laser which is not uniform but rather of a spiky nature shown in figure 4.9.

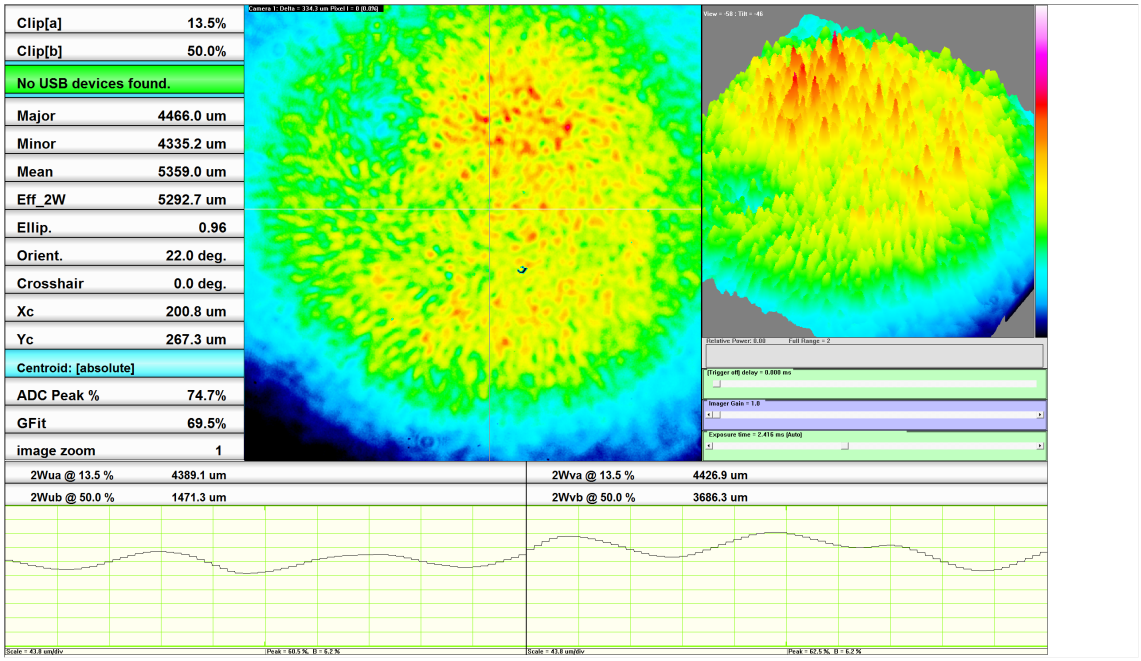

Figure 4.9: Laser beam profile

Although we do not make any further investigation on the velocity profile we estimated, we want to give an example where [42] used the time averaged velocity profile of a laser induced particle flow in a 2D complex plasma to calculate the viscosity of the system. There they calculated the kinematic viscosity by the shear stress σ and the shear rate $\dot{\gamma}$ of the 2D system.
According to [42] the shear rate can be calculated by the velocity profile of the complex plasma by

$$\dot{\gamma} = \frac{dv_x}{dx} \tag{4.3}$$

where x is the direction perpendicular to the particle flow. The shear stress can be calculated in polar coordinates for the 2D complex plasma by

$$\sigma = -\frac{1}{2}n^2 \int_0^{r_{max}} dr r^2 \frac{dV}{dr} \int_0^{2\pi} d\phi \cos(\phi)\sin(\phi) g(r,\phi) \tag{4.4}$$

where n denotes the areal particle number density, $V(r)$ the pair interaction potential and was approximated by the Yukawa potential and $g(r,\phi)$ the two dimensional pair correlation function.
By further calculating the areal mass density $\rho = m \cdot n$ of the system, where m is the mass of a single particle, one can calculate the kinematic viscosity ν by

$$\nu = \frac{\sigma}{\rho\dot{\gamma}} \tag{4.5}$$

This is an example where the precise estimation of a velocity profile is crucial for the calculation of an important physical parameter, the kinematic viscosity.

4.4 Velocity wave analysis

Finally, we show another application of the velocity estimation method. Besides laser induced flows with constant laser power we also have experimental data for particle flow with a modulated laser power of $(6 \pm 2)\ W$. The modulation frequencies are known to be 1, 3 and 10 Hz.
Here we again calculate the time evolution of a section in the middle of the images as in the previous section. The modulation of the laser intensity ideally creates a modulation of the particle velocities with the same frequency. If we estimate the mean velocity over a time, we can estimate this frequency of the velocity modulation. To achieve this, we calculate the power spectrum of the time evolution of the mean velocity value, which should have a distinct peak at the modulation frequency of the laser. To calculate the power spectrum of a discrete signal one has to first calculate the discrete Fourier transform of the signal, i.e. transform the signal from the time domain into the frequency domain.
The frequency dependent values X_f of the discrete Fourier transform can be calculated from the signal values in the time domain x_t of length N by following formula

$$\begin{aligned} X_f &= \sum_{t=0}^{N-1} x_t \exp(-2\pi i \frac{tf}{N}) \\ f &= 0, ..., N-1 \end{aligned} \tag{4.6}$$

The power spectrum can then be calculated by squaring the absolute value of every value of the Fourier transform.
In figures 4.10, 4.11 and 4.13 we show the time evolution of the mean velocity on the left and its power spectrum on the right. Like in the previous section we also fitted a linear function to the values starting at the twentieth frame. We afterwards subtract the fits from the original mean velocity signal to get a detrended signal from which the Fourier transform is calculated. In the insets in the plots of the time evolution we show the calculated fitting parameters and in the inset of the power spectrum the frequency of the maximum value and the frequency step. The width of the frequency step is solely dependent on the frame rate and the length of the signal and is independent of the accuracy of the velocity estimation of our method.

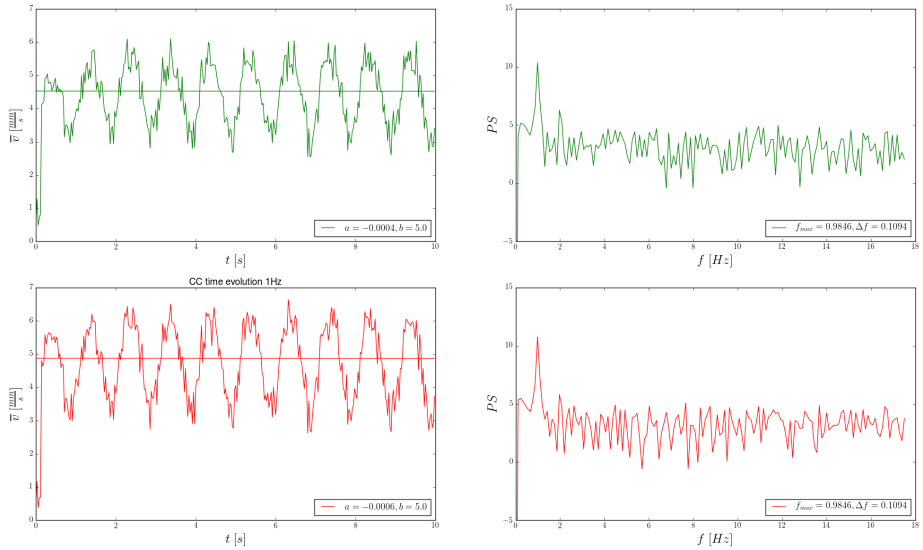

Figure 4.10: Frequency estimation of velocity wave driven by laser of $6 \pm 2W$ laser with frequency 1 Hz
left: time evolution of the mean velocity, right: power spectrum of the time evolution of the mean velocity

For a modulation frequency of one Hz the linear fit is nearly constant over time, with a slope in the order of at least 10^{-4}. The maximum value of the power spectrum appears at 0.9846 Hz for both similarity measures.

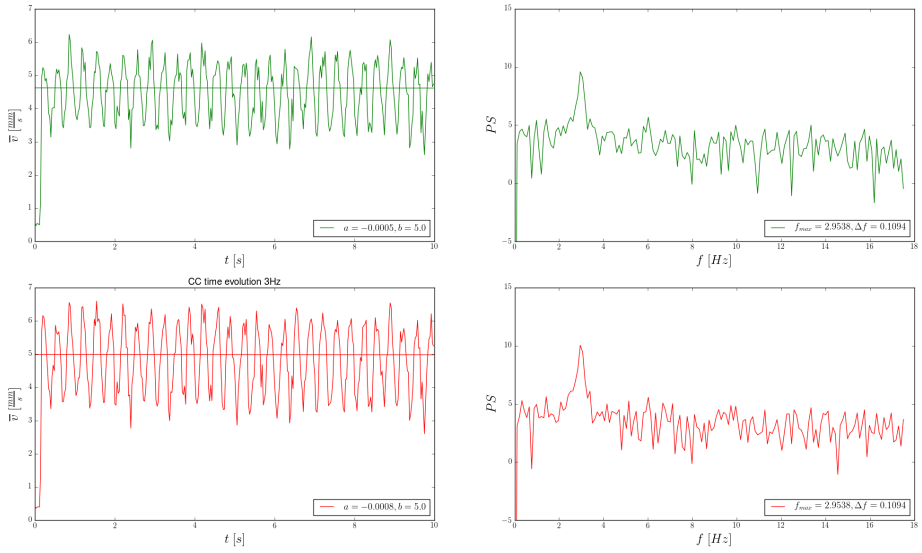

Figure 4.11: Frequency estimation of velocity wave driven by laser of $6 \pm 2W$ laser with frequency 3 Hz
left: time evolution of the mean velocity, right: power spectrum of the time evolution of the mean velocity

Like in the case of a modulation frequency of 1 Hz we observe a practically constant fit of the mean velocity value time evolution. Here the maximum value of the power spectrum appears at 2.9538 Hz in both cases.

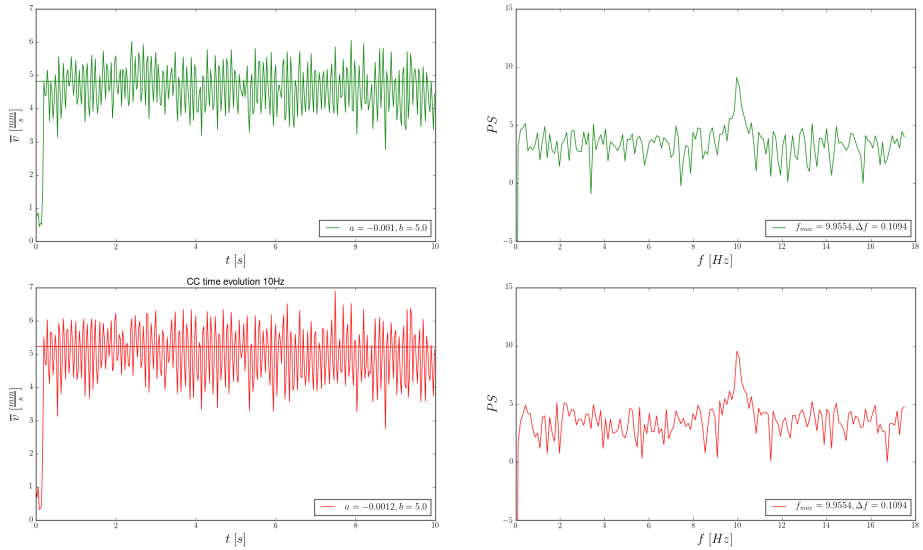

Figure 4.12: Velocity profile for PK4 data and minimum number of velocity values of 40

Figure 4.13: Frequency estimation of velocity wave driven by laser of $6 \pm 2W$ laser with frequency 10 Hz
left: time evolution of the mean velocity, right: power spectrum of the time evolution of the mean velocity

In the case of the modulation frequency of 10 Hz the slope of the fits m increases for both cases. Still they are both in the order of 10^{-3}. Here we calculate a maximum frequency of 9.9554 Hz.
Summarizing for all three modulation frequencies, we showed that the estimated modulation frequency of the mean particle velocity matches the modulation frequency of the laser, taking into account the uncertainty of the Fourier transform in frequency due to the finite signal length.
This results show a very good proof of concept for our velocity estimation method with real world data.

Chapter 5

Conclusions

We finally want to summarize the results we presented in this work.
Like [21] have shown, we found that the Pearsons Correlation Coefficient (CC) can be used to detect particles in the presence of Gaussian white noise and outperforms the Mutual information for high noise in this regime. However if the situation changes and one takes into account a signal dependent noise model, which is more suitable for a lot of imaging situations, we showed that the MI outperforms the CC in this noise regime by far for low SNR values. We also showed that in dense particle clouds, i.e. low inter-particle distances, the MI can separate nearby particles to a much greater extent than the CC. In contrast, in the case of anti-correlated regions, for example empty regions surrounded by particles, the MI can lead to false detections as it cannot distinguish between correlation and anti-correlation. But as we pointed out before, such a situation, including the positioning of four particles, is much more unlikely than two nearby particles. Therefore we highly suggest the MI for particle detection in the case of dense particle clouds.
Besides the position estimation we also proposed a new method to estimate the velocity of a particle out of a single image. We showed that the performance of the velocity estimation is highly dependent on the imaged size of the particles. For smaller particles we found that the MI is more accurate than the CC, for both the estimation of the velocity and for the estimation of the direction of motion. For increasing particle size the performance of both similarity measures was similar and decreased for large particles. After that we showed a typical application for our method where we estimated the velocity profile of a sequence of images of mid size particles and showed that we can recover the original velocity profile for low image noise precisely. However, for increasing noise we lost accuracy way faster than for example in the pure position or velocity estimation. Also we found that the estimation of the turbulent velocity profile is more complicated than the laminar velocity profile.
In the fourth section we presented the results of our method for experimental data coming from the PK-4 microgravity experiment on the International Space Station (ISS). We found some strong evidence for the superior performance of the MI for dense particle clouds. Furthermore, we showed the results for estimation of a velocity profile

of the PK-4 data, which resulted in reasonable results for both similarity measures. Besides the estimation of the velocity profile of a laser driven flow, we showed that we can correctly estimate the frequencies of velocity waves in the experimental data induced by a modulated laser power. This result is a excellent proof of concept for our velocity estimation method for real world data.
In this thesis we extended the method of particle detection by template matching, introduced by [21], by a nonlinear similarity measure, the MI, and a new template design. This new template design enables us to estimate not only the position of a particle, but also its velocity and direction of motion out of a single image. By this, we introduced a new method to highly increase the information available in a many particle system, whose time evolution is recorded by an optical device.

Chapter 6

Acknowledgements

First of all I want to thank my supervisors Prof. Dr. Gregor Morfill, Dr. Christoph Räth and the head of our group at the DLR Dr. Hubertus Thomas for giving me the opportunity for this master thesis. Especially I want to thank Dr. Räth for his constant support during my stay at the group. With his help I managed to overcome the complications I encountered in investigating this topic and without it I would not have accomplished the results I present here.
At this point I also want to thank all the other members in the group for their support. My special thanks go to Ingo Laut and Korbinian Schreiber for the interesting discussions we had and the pieces of advice they provided for all sorts of problems. I also want to thank Dr. Daniel Mohr for his support with my code and his patience with me when it comes to explaining all the details related to programming and computer sciences in general. I also want to express my gratitude to Dr. Christina Knapek, Dr. Volodymyr Nosenko, Dr. Peter Huber and Dr. Mikhail Pustylnik for their support in physical and technical questions. Thanks to Dr. Thomas I extended my stay at the group for another three months and I am really looking forward to it.

Bibliography

[1] H. M. Thomas and G. E. Morfill, "Melting dynamics of a plasma crystal," *Nature*, vol. 379, pp. 806–809, 02 1996.

[2] F. Melandsø, Å. Bjerkmo, G. Morfill, H. Thomas, and M. Zuzic, "Detection of stochastic waves in plasma monolayer crystals from video images," *Physics of Plasmas (1994-present)*, vol. 7, no. 11, pp. 4368–4378, 2000.

[3] V. Hadziavdic, F. Melandsø, and A. Hanssen, "Particle tracking from image sequences of complex plasma crystals," *Physics of Plasmas (1994-present)*, vol. 13, no. 5, p. 053504, 2006.

[4] A. Melzer, A. Homann, and A. Piel, "Experimental investigation of the melting transition of the plasma crystal," *Phys. Rev. E*, vol. 53, pp. 2757–2766, Mar 1996.

[5] D. B. Murphy, *Fundamentals of light microscopy and electronic imaging*. John Wiley & Sons, 2002.

[6] M. Zeng, J. Li, and Z. Peng, "The design of top-hat morphological filter and application to infrared target detection," *Infrared Physics & Technology*, vol. 48, no. 1, pp. 67–76, 2006.

[7] L. Yang, J. Yang, and K. Yang, "Adaptive detection for infrared small target under sea-sky complex background," *Electronics Letters*, vol. 40, no. 17, pp. 1083–1085, 2004.

[8] D. Gerlich and J. Ellenberg, "4d imaging to assay complex dynamics in live specimens.," *Nature cell biology*, pp. S14–9, 2003.

[9] E. Meijering, I. Smal, and G. Danuser, "Tracking in molecular bioimaging," *Signal Processing Magazine, IEEE*, vol. 23, no. 3, pp. 46–53, 2006.

[10] S. S. Rogers, T. A. Waigh, X. Zhao, and J. R. Lu, "Precise particle tracking against a complicated background: polynomial fitting with gaussian weight," *Physical Biology*, vol. 4, no. 3, p. 220, 2007.

[11] D. Sage, F. R. Neumann, F. Hediger, S. M. Gasser, and M. Unser, "Automatic tracking of individual fluorescence particles: application to the study of chromosome dynamics," *Image Processing, IEEE Transactions on*, vol. 14, no. 9, pp. 1372–1383, 2005.

[12] B. Shuang, J. Chen, L. Kisley, and C. F. Landes, "Troika of single particle tracking programing: Snr enhancement, particle identification, and mapping," *Physical Chemistry Chemical Physics*, vol. 16, no. 2, pp. 624–634, 2014.

[13] J. C. Crocker and D. G. Grier, "Methods of digital video microscopy for colloidal studies," *Journal of colloid and interface science*, vol. 179, no. 1, pp. 298–310, 1996.

[14] K. Celler, G. P. van Wezel, and J. Willemse, "Single particle tracking of dynamically localizing tata complexes in streptomyces coelicolor," *Biochemical and Biophysical Research Communications*, vol. 438, no. 1, pp. 38 – 42, 2013.

[15] T.-C. Ku, Y.-N. Huang, C.-C. Huang, D.-M. Yang, L.-S. Kao, T.-Y. Chiu, C.-F. Hsieh, P.-Y. Wu, Y.-S. Tsai, and C.-C. Lin, "An automated tracking system to measure the dynamic properties of vesicles in living cells," *Microscopy Research and Technique*, vol. 70, no. 2, pp. 119–134, 2007.

[16] I. Izeddin, J. Boulanger, V. Racine, C. Specht, A. Kechkar, D. Nair, A. Triller, D. Choquet, M. Dahan, and J. Sibarita, "Wavelet analysis for single molecule localization microscopy," *Optics express*, vol. 20, no. 3, pp. 2081–2095, 2012.

[17] N. Chenouard, I. Bloch, and J.-C. Olivo-Marin, "Multiple hypothesis tracking in microscopy images," in *Biomedical Imaging: From Nano to Macro, 2009. ISBI '09. IEEE International Symposium on*, pp. 1346–1349, June 2009.

[18] I. Smal, M. Loog, W. Niessen, and E. Meijering, "Quantitative comparison of spot detection methods in fluorescence microscopy," *Medical Imaging, IEEE Transactions on*, vol. 29, no. 2, pp. 282–301, 2010.

[19] L. G. Brown, "A survey of image registration techniques," *ACM computing surveys (CSUR)*, vol. 24, no. 4, pp. 325–376, 1992.

[20] B. Zitova and J. Flusser, "Image registration methods: a survey," *Image and vision computing*, vol. 21, no. 11, pp. 977–1000, 2003.

[21] Y. Gao and M. E. Helgeson, "Texture analysis microscopy: quantifying structure in low-fidelity images of dense fluids," *Optics express*, vol. 22, no. 8, pp. 10046–10063, 2014.

[22] J. Lee Rodgers and W. A. Nicewander, "Thirteen ways to look at the correlation coefficient," *The American Statistician*, vol. 42, no. 1, pp. 59–66, 1988.

[23] R. Taylor, "Interpretation of the correlation coefficient: a basic review," *Journal of diagnostic medical sonography*, vol. 6, no. 1, pp. 35–39, 1990.

[24] C. E. Shannon, "A mathematical theory of communication," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 1, pp. 3–55, 2001.

[25] A. Collignon, F. Maes, D. Delaere, D. Vandermeulen, P. Suetens, and G. Marchal, "Automated multi-modality image registration based on information theory," *In: Bizais*, 1995.

[26] P. Viola and W. Wells, "Alignment by maximization of mutual information," in *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pp. 16–23, Jun 1995.

[27] R. Steuer, J. Kurths, C. O. Daub, J. Weise, and J. Selbig, "The mutual information: detecting and evaluating dependencies between variables," *Bioinformatics*, vol. 18, no. suppl 2, pp. S231–S240, 2002.

[28] N. D. Dowson, R. Bowden, and T. Kadir, "Image template matching using mutual information and np-windows," in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, vol. 2, pp. 1186–1191, IEEE, 2006.

[29] J. P. Pluim, J. A. Maintz, M. Viergever, *et al.*, "Mutual-information-based registration of medical images: a survey," *Medical Imaging, IEEE Transactions on*, vol. 22, no. 8, pp. 986–1004, 2003.

[30] J. C. Curlander and R. N. McDonough, *Synthetic aperture radar*. John Wiley & Sons, 1991.

[31] C. B. Burckhardt, "Speckle in ultrasound b-mode scans," *Sonics and Ultrasonics, IEEE Transactions on*, vol. 25, no. 1, pp. 1–6, 1978.

[32] H. Lu, I.-T. Hsiao, X. Li, and Z. Liang, "Noise properties of low-dose ct projections and noise treatment by scale transformations," in *Nuclear Science Symposium Conference Record, 2001 IEEE*, vol. 3, pp. 1662–1666, IEEE, 2001.

[33] H. Gudbjartsson and S. Patz, "The rician distribution of noisy mri data," *Magnetic resonance in medicine*, vol. 34, no. 6, pp. 910–914, 1995.

[34] A. J. Blanksby, M. J. Loinaz, D. Inglis, and B. D. Ackland, "Noise performance of a color cmos photogate image sensor," in *Electron Devices Meeting, 1997. IEDM'97. Technical Digest., International*, pp. 205–208, IEEE, 1997.

[35] G. E. Healey and R. Kondepudy, "Radiometric ccd camera calibration and noise estimation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 16, no. 3, pp. 267–276, 1994.

[36] A. K. Jain, *Fundamentals of digital image processing*. Prentice-Hall, Inc., 1989.

[37] I. Grant, "Particle image velocimetry: a review," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 211, no. 1, pp. 55–76, 1997.

[38] R. J. Adrian, "Twenty years of particle image velocimetry," *Experiments in fluids*, vol. 39, no. 2, pp. 159–169, 2005.

[39] S. Blackman, "Multiple hypothesis tracking for multiple target tracking," *Aerospace and Electronic Systems Magazine, IEEE*, vol. 19, pp. 5–18, Jan 2004.

[40] N. Chenouard, I. Smal, F. De Chaumont, M. Maška, I. F. Sbalzarini, Y. Gong, J. Cardinale, C. Carthel, S. Coraluppi, M. Winter, *et al.*, "Objective comparison of particle tracking methods," *Nature methods*, vol. 11, no. 3, pp. 281–289, 2014.

[41] V. Nosenko and J. Goree, "Shear flows and shear viscosity in a two-dimensional yukawa system (dusty plasma)," *Physical review letters*, vol. 93, no. 15, p. 155004, 2004.

[42] V. Nosenko, A. Ivlev, and G. Morfill, "Anisotropic shear melting and recrystallization of a two-dimensional complex plasma," *Physical Review E*, vol. 87, no. 4, p. 043115, 2013.

[43] S. Kullback and R. A. Leibler, "On information and sufficiency," *Ann. Math. Statist.*, vol. 22, pp. 79–86, 03 1951.

[44] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens, "Multimodality image registration by maximization of mutual information," *Medical Imaging, IEEE Transactions on*, vol. 16, no. 2, pp. 187–198, 1997.

[45] J. Walters-Williams and Y. Li, "Estimation of mutual information: A survey," in *Rough Sets and Knowledge Technology*, pp. 389–396, Springer, 2009.

[46] A. Strehl and J. Ghosh, "Cluster ensembles—a knowledge reuse framework for combining multiple partitions," *The Journal of Machine Learning Research*, vol. 3, pp. 583–617, 2003.

[47] F. White, *Fluid mechanics*. McGraw-Hill Higher Education, 2015.

[48] V. Fortov, G. Morfill, O. Petrov, M. Thoma, A. Usachev, H. Hoefner, A. Zobnin, M. Kretschmer, S. Ratynskaia, M. Fink, *et al.*, "The project'plasmakristall-4'(pk-4)a new stage in investigations of dusty plasmas under microgravity conditions: first results and future plans," *Plasma physics and controlled fusion*, vol. 47, no. 12B, p. B537, 2005.

[49] M. H. Thoma, M. Fink, H. Höfner, M. Kretschmer, S. Khrapak, S. V. Ratynskaia, V. V. Yaroshenko, G. E. Morfill, O. F. Petrov, A. D. Usachev, *et al.*, "Pk-4: Complex plasmas in spacethe next generation," *Plasma Science, IEEE Transactions on*, vol. 35, no. 2, pp. 255–259, 2007.

[50] M. A. Fink, M. H. Thoma, and G. E. Morfill, "Pk-4 science activities in microgravity," *Microgravity Science and Technology*, vol. 23, no. 2, pp. 169–171, 2011.

Erklärung

Hiermit erkläre ich, die vorliegende Arbeit selbständig verfasst zu haben und keine anderen als die in der Arbeit angegebenen Quellen und Hilfsmittel benutzt zu haben.

Ort, Datum

Unterschrift